

Chapter 2

Supervised Texture Classification for Intravascular Tissue Characterization

Oriol Pujol¹ and Petia Radeva¹

2.1 Introduction

Vascular disease, stroke, and arterial dissection or rupture of coronary arteries are considered some of the main causes of mortality in present days. The behavior of the atherosclerotic lesions depends not only on the degree of lumen narrowing but also on the histological composition that causes that narrowing. Therefore, tissue characterization is a fundamental tool for studying and diagnosing the pathologies and lesions associated to the vascular tree.

Although important, tissue characterization is an arduous task that requires manual identification by specialists of the tissues and proper tissue visualization. Intravascular ultrasound (IVUS) imaging is a well suited visualization technique for such task as it provides a cross-sectional cut of the coronary vessel, unveiling its histological properties and tissue organization.

IVUS is a widespread technique accepted in clinical practice to fill up the lack of information provided by classical coronary angiography on vessel morphology. It has a prominent role evaluating the artery lesion after a interventional coronary procedure such as balloon dilation of the vessel, stent implantation, laser angioplasty, or atherectomy.

IVUS displays the morphology and histological properties of a cross section of a vessel [1]. Figure 2.1 shows a good example of different tissues in an

¹Computer Vision Center, Universitat Autònoma de Barcelona, Campus UAB, Edifici O, 08193 Bellaterra (Barcelona), Spain

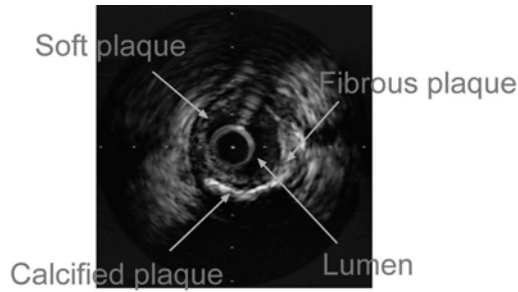


Figure 2.1: Typical IVUS image presenting different kind of tissues.

IVUS image. It is generally accepted that the different kind of plaque tissues distinguishable in IVUS images is threefold: *Calcium formation* is characterized by a very high echorefectivity and absorbtion of the emitted pulse from the transducer. This behavior produces a deep shadowing effect behind calcium plaques. In the figure, calcium formation can be seen at three o'clock and from five to seven o'clock. *Fibrous plaque* has medium echorefectivity resembling that of the adventitia. This tissue has a good transmission coefficient allowing the pulse to travel through the tissue, and therefore, providing a wider range of visualization. This kind of tissue can be observed from three o'clock to five o'clock. *Soft plaque* or *Fibro-Fatty plaque* is the less echoreflective of the three kind of tissues. It also has good transmission coefficient allowing to see what is behind this kind of plaque. Observing the figure, a soft plaque configuration is displayed from seven o'clock to three o'clock.

Because of time consumption and subjectivity of the classification depending on the specialist, there is a crescent interest of the medical community in developing automatic tissue characterization procedures. This is accentuated because the procedure for tissue classification by physicians implies the manual analysis of IVUS images.

The problem of automatic tissue characterization has been widely studied in different medical fields. The unreliability of gray-level only methods to achieve good discrimination among the different kind of tissues forces us to use more complex measures, usually based on texture analysis. Texture analysis has played a prominent role in computer vision to solve tissue characterization problems in medical imaging [2–9].

Several researching groups have reported different approximations to characterize the tissue of IVUS image.

Vandenberg in [10] base their contribution on reducing the noise of the image to have a clear representation of the tissue. The noise reduction is achieved by averaging sets of images when the least variance in diameter of the IVUS occurs. At the end, a fuzzy logic based expert is set to discriminate among the tissues.

Nailon and McLaughlin devote several efforts to IVUS tissue characterization. In [11] they use classic Haralick texture statistics to discriminate among tissues. In [12] the authors propose the use of co-occurrence matrices texture analysis and fractal texture analysis to characterize intravascular tissue. Thirteen features plus fractal dimension derived from Brownian motion are used for this task. The conclusion shows that fractal dimension is unable to discriminate between calcium and fibrous plaque but helps in fibrous versus lipidic plaque. On the other hand, co-occurrence matrices are well suited for the overall classification. In [13], it is stated that the discriminative power of fractal dimension is poor when trying to separate fibrotic tissue, lipidic tissue, and foam cells. The method used is based on fractal dimension estimation techniques (box-counting, brownian motion, and frequency domain).

Spencer in [14] center their work on spectral analysis. Different features are compared: mean power, maximum power, spectral slope, and 0 Hz interception. The work concludes with the 0 Hz spectral slope as the most discriminative feature.

Dixon in [15] use co-occurrence matrices and discriminant analysis to evaluate the different kind of tissues in IVUS images.

Ahmed and Leyman in [16] use a radial transform and correlation for pattern matching. The features used are higher order statistics such as kurtosis, skewness, and up to four order cumulants. The results provided appear to have fairly good visual recognition rate.

The work of de Korte and van der Steen [17] opens a new proposal based on assessing the local strain of the atherosclerotic vessel wall to identify different plaque components. This very promising technique, called *elastography*, is based on estimating the radial strain by performing cross-correlation analysis on pairs of IVUS at a certain intracoronary pressure.

Probably, one of the most interesting work in this field is the one provided by Zhang and Sonka in [18]. This work is much more complex trying to evaluate the full morphology of the vessel. Detecting the plaque and adventitia borders and characterizing the different kind of tissues, the tissue discrimination is done using a combination of well-known techniques previously reported in the

literature, co-occurrence matrices and fractal dimension from brownian motion, and adding two more strategies to the amalgam of features: run-length measures and radial profile. The experiments assess the accuracy of the method quantitatively.

Most of the literature found in the tissue characterization matters use texture features, co-occurrence matrices being the most popular of all feature extractors. Further work has been done trying to use other kind of texture feature extractors and IVUS images, and although not specifically centered on tissue characterization, the usage of different texture features in plaque border assessment is reported, which can be easily extrapolated to tissue characterization. In [19], derivatives of Gaussian; wavelets, co-occurrence matrices, Gabor filters, and accumulation local moments are evaluated and used to classify blood from plaque. The work highlights the discriminative power of co-occurrence matrices derivatives of Gaussian and accumulation local moments. Other works such as [20] provide some hints on how to achieve a fast framework based on local binary patterns and fast high-performance classifiers. This last line of investigation overcomes one of the most significant drawbacks of the texture based tissue characterization systems, the speed. Texture descriptors are inherently slow to be computed. With the proposal of the feature extractor based on local binary patterns a good discriminative power is ensured as well as a fast technique for tissue characterization.

Whatever method we use in the tissue characterization task, we follow an underlying main methodology. First, we need to extract some features describing the tissue variations. This first step is critical since the features chosen have to be able to describe each kind of tissue in a unique way so that it cannot be confused with another one. In this category of feature extraction we should consider the *co-occurrence matrix measures*, *local binary patterns*, etc. The second step is the classification of the extracted features. Depending on the complexity of the feature data some methods will fit better than others. In most cases, high-dimensional spaces are generated, so we should consider the use of dimensionality reduction methods such as *principal component analysis* or *Fisher linear discriminant analysis*. Either a dimensionality reduction process is needed or not, this step will require a classification procedure. This procedure can be supervised, if we provide samples of each tissue to be classified so that the system “knows” *a priori* what the tissues are, or unsupervised, if we allow the system to try to find which are the different kind of tissues by itself. In

this category, we can find *clustering* methods for unsupervised classification and, for supervised classification, methods like *maximum likelihood*, *nearest neighbors*, etc.

The following sections are devoted to describe the following: First, the most significative texture methods used in the literature. Secondly, some of the most successful classification methods applied to IVUS characterization are reviewed. Third, we describe the result of using such techniques for tissue characterization and conclude about the optimal feature space to describe tissue plaque and the best classifier to discriminate it.

2.2 Feature Spaces

Gray-level thresholding is not enough for robust tissue characterization. Therefore, it is generally approached as a texture discrimination problem. This line of work is a classical extension of previous works on biological characterization, which also relies on texture features as has been mentioned in the former section. The co-occurrence matrix is the most favored and well known of the texture feature extraction methods due to its discriminative power in this particular problem but it is not the only one nor the fastest method available. In this section, we make a review of different texture methods that can be applied to the problem in particular, from the *co-occurrence matrix measures* method to the most recent texture feature extractor, *local binary patterns*.

To illustrate the texture feature extraction process we have selected a set of techniques basing our criterion of selection on the most widespread methods for tissue characterization and the most discriminative feature extractors reported in the literature [21].

Basically, the different methods of feature extraction emphasize on different fundamental properties of the texture such as scale, statistics, or structure. In this way, under the nonelemental statistics property we can find two well-known techniques, co-occurrence methods [22] and higher order statistics represented by moments [23]. Under the label of scale property we should mention methods such as derivatives of Gaussian [24], Gabor filters [25], or wavelet techniques [26]. Regarding structure-related measures there are methods such as fractal dimension [27] and local binary patterns [28].

To introduce the texture feature extraction methods we divide them into two groups: The first group, that forms the *statistic-related methods*, is comprised of co-occurrence matrix measures, accumulation local moments, fractal dimension, and local binary patterns. All these methods are somehow related to statistics. Co-occurrence matrix measures are second-order measures associated to the probability density function estimation provided by the co-occurrence matrix. Accumulation local moments are directly related to statistics. Fractal dimension is an approximation of the roughness of a texture. Local binary patterns provides a measure of the local inhomogeneity based on an “averaging” process. The second group, that forms the *analytic kernel-based extraction techniques*, comprises Gabor bank of filters, derivatives of Gaussian filters, and wavelet decomposition. The last three methods are derived from analytic functions and sampled to form a set of filters, each focused on the extraction of a certain feature.

2.2.1 Statistic-Related Methods

2.2.1.1 Co-occurrence Matrix Approach

In 1962 Julesz [29] showed the importance of texture segregation using second-order statistics. Since then, different tools have been used to exploit this issue. The gray-level co-occurrence matrix is a well-known statistical tool for extracting second-order texture information from images [22]. In the co-occurrence method, the relative frequencies of gray-level pairs of pixels at certain relative displacement are computed and sorted in a matrix, the *co-occurrence matrix* \mathbf{P} . The co-occurrence matrix can be thought of as an estimate of the joint probability density function of gray-level pairs in an image. For G gray levels in the image, \mathbf{P} will be of size $G \times G$. If G is large, the number of pixel pairs contributing to each element, $p_{i,j}$ in \mathbf{P} is low, and the statistical significance poor. On the other hand, if the number of gray levels is low, much of the texture information may be lost in the image quantization. The element values in the matrix, when normalized, are bounded by $[0, 1]$, and the sum of all element values is equal to 1.

$$P(i, j, D, \theta) = P(I(l, m) = i \quad \text{and} \quad I(l + D \cos(\theta), m + D \sin(\theta)) = j$$

where $I(l, m)$ is the image at pixel (l, m) , D is the distance between pixels,

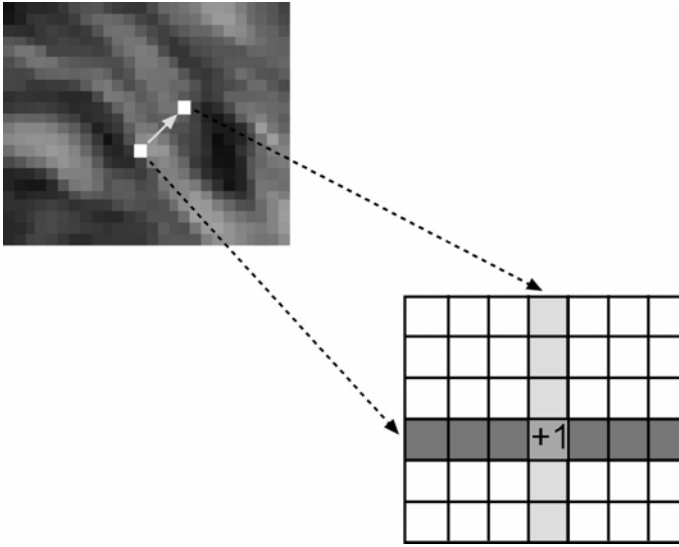


Figure 2.2: Co-occurrence matrix explanation diagram (see text).

and θ is the angle. It has been proved by other researchers [21, 30] that the nearest neighbor pairs at distance D at orientations $\theta = \{0^\circ, 45^\circ, 90^\circ, 135^\circ\}$ are the minimum set needed to describe the texture second-order statistic measures. Figure 2.2 illustrates the method providing a graphical explanation. The main idea is to create a “histogram” of the occurrences of having two pixels of certain gray levels at a determined distance with a fixed angle. Practically, we add one to the cell of the matrix pointed by the gray levels of two pixels (one pixel gray level gives the file and the other the column of the matrix) that fulfill the requirement of being at a certain predefined distance and angle.

Once the matrix is computed several characterizing measures are extracted. Many of these features are derived by weighting each of the matrix element values and then summing these weighted values to form the feature value. The weight applied to each element is based on a feature-weighting function, so by varying this function, different texture information can be extracted from the matrix. We present here some of the most important measures that characterize the co-occurrence matrices: energy, entropy, inverse difference moment, shade, inertia, and promenance [30]. Let us introduce some notation for the definition of the features:

$P(i, j)$ is the (i, j) th element of a normalized co-occurrence matrix

$$\begin{aligned}
 P_x(i) &= \sum_j P(i, j) \\
 P_y(j) &= \sum_i P(i, j) \\
 \mu_x &= \sum_i i \sum_j P(i, j) = \sum_i iP_x(i) = E\{i\} \\
 \mu_y &= \sum_j j \sum_i P(i, j) = \sum_j jP_y(j) = E\{j\}
 \end{aligned}$$

With the above notation, the features can be written as follows:

$$\begin{aligned}
 \text{Energy} &= \sum_{i,j} P(i, j)^2 \\
 \text{Entropy} &= - \sum_{i,j} P(i, j) \log P(i, j) \\
 \text{Inverse difference moment} &= \sum_{i,j} \frac{1}{1 + (i - j)^2} P(i, j) \\
 \text{Shade} &= \sum_{i,j} (i + j - \mu_x - \mu_y)^3 P(i, j) \\
 \text{Inertia} &= \sum_{i,j} (i - j)^2 P(i, j) \\
 \text{Promenance} &= \sum_{i,j} (i + j - \mu_x - \mu_y)^4 P(i, j)
 \end{aligned}$$

Hence, we create a *feature vector* for each of the pixels by assigning each feature measure to a component of the feature vector. Given that we have four different orientations and the six measures for each orientation, the feature vector is a 24-dimensional vector for each pixel and for each distance. Since we have used two distances $D = 2$ and $D = 3$, the final vector is a 48-dimensional vector.

Figure 2.3 shows responses for different measures on the co-occurrence matrices. Although a straightforward interpretation of the feature extraction response is not easy, some deduction can be made by observing the figures. Figure 2.3(b) shows shade measure; as its name indicates it is related to the shadowed areas in the image, and thus, localizing the shadowing behind the calcium plaque. Figure 2.3(c) shows inverse different moment response, this measure seems to be related to the first derivative of the image, enhancing contours. Figure 2.3(d) depicts the output for the inertia measure, which seems to have some relationship with local homogeneity of the image.

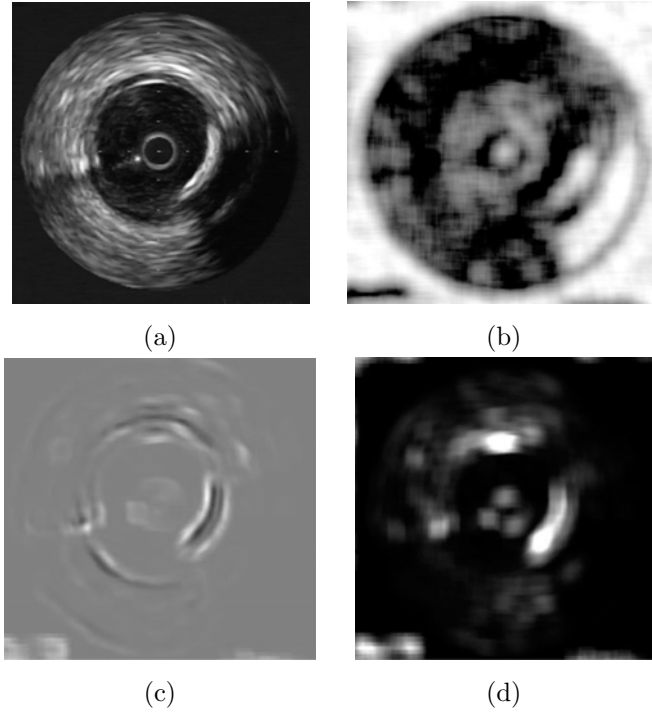


Figure 2.3: Response of an IVUS image to different measures of the co-occurrence matrix. (a) Original image, (b) measure shade response, (c) inverse different moment, and (d) inertia.

2.2.1.2 Accumulation Local Moments

Geometric moments have been used effectively for texture segmentation in many different application domains [23]. In addition, other kind of moments have been proposed: Zernique moments, Legendre moments, etc. By definition, any set of parameters obtained by projecting an image onto a two-dimensional polynomial basis is called moments. Then, since different sets of polynomials up to the same order define the same subspace, any complete set of moments up to given order can be obtained from any other set of moments up to the same order. The computation of some of these sets of moments leads to very long processing times, so in this section a particular fast computed moment set has been chosen. This set of moments is known as the accumulation local moments. Two kind of accumulation local moments can be computed, direct accumulation and reverse accumulation. Since direct accumulation is more sensitive to round

off errors and small perturbations in the input data [31], the reverse accumulation moments are recommendable.

The reverse accumulation moment of order $(k - 1, l - 1)$ of matrix \mathbf{I}_{ab} is the value of $\mathbf{I}_{ab}[1, 1]$ after bottom-up accumulating its column k times (i.e., after applying k times the assignment $\mathbf{I}_{ab}[a - i, j] \leftarrow \mathbf{I}_{ab}[a - i, j] + \mathbf{I}_{ab}[a - i + 1, j]$, for $i = 0$ to $a - 1$, and for $j = 1$ to b), and accumulating the resulting first row from right to left l times (i.e., after applying l times the assignment $\mathbf{I}_{ab}[1, b - j] \leftarrow \mathbf{I}_{ab}[1, b - j] + \mathbf{I}_{ab}[1, b - j + 1]$, for $j = 1$ to $b - 1$). The reverse accumulation moment matrix is defined so that $\mathbf{R}_{mn}[k, l]$ is the reverse accumulation moment of order $(k - 1, l - 1)$.

Consider the matrix in the following example:

$$\begin{pmatrix} 0 & 1 & 2 \\ 1 & 1 & 1 \\ 4 & 2 & 3 \end{pmatrix}$$

According to the definition, its reverse accumulation moment of order (1,2) requires two column accumulations,

$$\begin{pmatrix} 5 & 4 & 6 \\ 5 & 3 & 4 \\ 4 & 2 & 3 \end{pmatrix} \rightarrow \begin{pmatrix} 14 & 9 & 13 \\ 9 & 5 & 7 \\ 4 & 2 & 3 \end{pmatrix}$$

and three right to left accumulations of the first row:

$$\begin{pmatrix} 36 & 22 & 13 \\ 9 & 5 & 7 \\ 4 & 2 & 3 \end{pmatrix} \rightarrow \begin{pmatrix} 71 & 35 & 13 \\ 9 & 5 & 7 \\ 4 & 2 & 3 \end{pmatrix} \rightarrow \begin{pmatrix} 119 & 48 & 13 \\ 9 & 5 & 7 \\ 4 & 2 & 3 \end{pmatrix}$$

Then it is said that the reverse accumulation moment of order (1,2) of the former matrix is 119.

The set of moments alone is not sufficient to obtain good texture features in certain images. Some iso-second order texture pairs, which are preattentively discriminable by humans, would have the same average energy over finite regions. However, their distribution would be different for the different textures. One solution suggested by Caelli is to introduce a nonlinear transducer that maps moments to texture features [32]. Several functions have been proposed in the literature: logistic, sigmoidal, power function, or absolute deviation of feature vectors from the mean [23]. The function we have chosen is the hyperbolic tangent function, which is logistic in shape. Using the accumulation moments image I_m and a nonlinear operator $|\tanh(\sigma(I_m - \bar{I}_m))|$ an “average” is performed

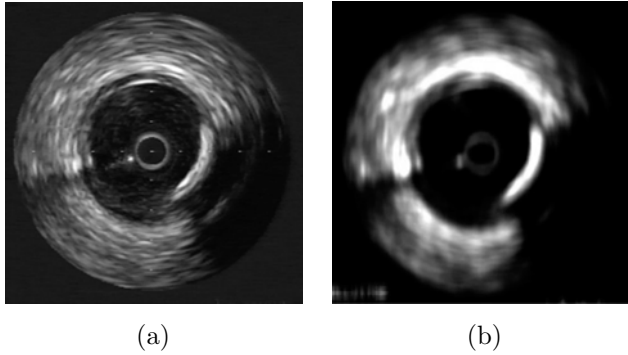


Figure 2.4: Accumulation local moments response. (a) Original image. (b) Accumulation local moment of order (3,1).

throughout the region of interest. The parameter σ controls the shape of the logistic function. Therefore each textural feature will be the result of the application of the nonlinear operator to the computed moments. If $n = k \cdot l$ moments are computed over the image, then the dimension of the feature vector will be n . Hence, a n -dimensional point is associated with each pixel of the image.

Figure 2.4 shows the response of moment (3,1) on an IVUS image. In this figure, the response seems to have a smoothing and enhancing effect, clearly resembling diffusion techniques.

2.2.1.3 Fractal Analysis

Another classic tool for texture description is the fractal analysis [13, 33], characterized by the fractal dimension. We talk roughly about fractal structures when a geometric shape can be subdivided in parts, each of which are approximately a reduced copy of the whole (this property is also referred as self-similarity). The introduction of fractals by Mandelbrot [27] allowed a characterization of complex structures that could not be described by a single measure using Euclidean geometry. This measure is the *fractal dimension*, which is related to the degree of irregularity of the surface texture.

The fractal structures can be divided into two subclasses: the deterministic fractals and the random fractals. Deterministic fractals are strictly self-similar, that is, they appear identical over a range of magnification scales. On the other hand, random fractals are statistical self-similar. The similarity between two scales of the fractal is ruled by a statistical relationship.

The fractal dimension represents the disorder of an object. The higher the dimension, the more complex the object is. Contrary to the Euclidian dimension, the fractal dimension is not constrained to integer dimensions.

The concept of fractals can be easily extrapolated to image analysis if we consider the image as a three-dimensional surface in which the height at each point is given by the gray value of the pixel.

Different approaches have been proposed to compute the fractal dimension of an object. Herein we consider only three classical approaches: box-counting, Brownian motion, and Fourier analysis.

Box-Counting. The box-counting method is an approximation to the fractal dimension as it is conceptually related to self-similarity.

In this method the object to be evaluated is placed on a square mesh of various sizes, r . The number of mesh boxes, N , that contain any part of the fractal structure are counted.

It has been proved that in a self-similar structures there is a relationship between the reduction factor r and the number of divisions N into which the structure can be divided:

$$Nr^D = 1$$

where D is the self-similarity dimension. Therefore, the fractal dimension can be easily written as

$$D = \frac{\log N}{\log 1/r}$$

This process is done at various scales by altering the square size r . Therefore, the box-counting dimension is the slope of the regression line that better approximates the data on the plot produced by $\log N \times \log 1/r$.

Fractal Dimension from Brownian Motion. The fractal dimension is found by considering the absolute intensity difference of pixel pairs, $I(p_1) - I(p_2)$, at different scales. It can be shown that for a fractal Brownian surface the following relationship must be satisfied:

$$E(|I(p_1) - I(p_2)|) \propto (\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2})^H$$

where E is the mean and H the Hurst coefficient. The fractal dimension is related to H in the following way: $D = 3 - H$. In the same way than the former

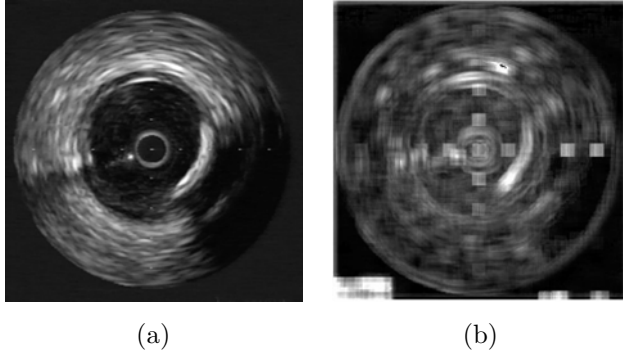


Figure 2.5: Fractal dimension from box-counting response. (a) Original image. (b) Fractal dimension response with neighborhoods of 10×10 .

method for calculating the fractal dimension the mean difference of intensities is calculated for different scales (each scale given by the euclidian distance between two pixels), and the slope of the regression line between $\log E(|I(p_1) - I(p_2)|)$ and $\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$ gives the Hurst parameter.

Triangular Prism Surface Area Method. The triangular prism surface area (TPSA) algorithm considers an approximation of the “area” of the fractal structure using triangular prisms. If a rectangular neighborhood is defined by its vertices A, B, C, and D, the area of this neighborhood is calculated by tessellating the surface with four triangles defined for each consecutive vertex and the center of the neighborhood.

The area of all triangles for every central pixel is summed up to the entire area for different scales. The double logarithmic Richardson–Mandelbrot plot should again yield a linear line whose slope is used to determine the TPSA dimension. Figure 2.5 shows the fractal dimension value of each pixel of an IVUS image considering the fractal dimension of a neighborhood around the pixel. The size of the neighborhood is 10×10 . The response of this technique seems to take into account the border information of the structures in the image.

2.2.1.4 Local Binary Patterns

Local binary patterns [28] are a feature extraction operator used for detecting “uniform” local binary patterns at circular neighborhoods of any quantization of

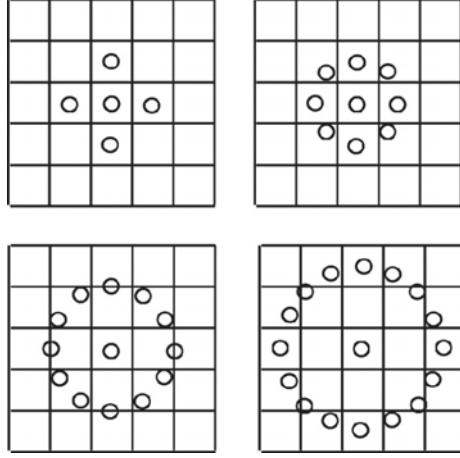


Figure 2.6: Typical neighbors: (Top left) $P = 4$, $R = 1.0$; (top right) $P = 8$, $R = 1.0$; (bottom left) $P = 12$, $R = 1.5$; (bottom right) $P = 16$, $R = 2.0$.

the angular space and at any spatial resolution. The operator is derived based on a circularly symmetric neighbor set of P members on a circle of radius R . It is denoted by $LBP_{P,R}^{iu2}$. Parameter P controls the quantization of the angular space, and R determines the spatial resolution of the operator. Figure 2.6 shows typical neighborhood sets. To achieve gray-scale invariance, the gray value of the center pixel (g_c) is subtracted from the gray values of the circularly symmetric neighborhood g_p ($p = 0, 1, \dots, P - 1$) and assigned a value of 1 if the difference is positive and 0 if negative.

$$s(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

By assigning a binomial factor 2^p for each value obtained, we transform the neighborhood into a single value. This value is the $LBP_{R,P}$:

$$LBP_{R,P} = \sum_{p=0}^P s(g_p - g_c) \cdot 2^p$$

To achieve rotation invariance the pattern set is rotated as many times as necessary to achieve a maximal number of the most significant bits, starting always from the same pixel. The last stage of the operator consists on keeping the information of “uniform” patterns while filtering the rest. This is achieved using a transition count function U . U is a function that counts the number of transitions

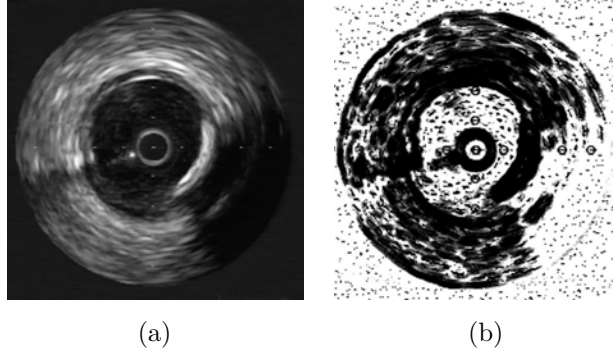


Figure 2.7: Local binary pattern response. (a) Original image. (b) Local binary pattern output with parameters $R = 3$, $P = 24$.

0/1, 1/0 while we move over the neighborhood:

$$U(LBP_{P,R}) = |s(g_{P-1} - g_c) - s(g_0 - g_c)| + \sum_{p=1}^{P-1} |s(g_p - g_c) - s(g_{p-1} - g_c)|$$

Therefore,

$$LBP_{P,R}^{riu2} = \begin{cases} LBP_{P,R}^{ri} & \text{if } U(LBP_{P,R}) \leq 2 \\ P + 1 & \text{otherwise.} \end{cases}$$

Figure 2.7 shows an example of an IVUS image filtered using a uniform rotation invariant local binary pattern with values $P = 24$, $R = 3$. The feature extraction image displayed in the figure looks like a discrete response focussed on the structure shape and homogeneity.

2.2.2 Analytic Kernel-Based Methods

2.2.2.1 Derivatives of Gaussian

In order to handle image structures at different scales in a consistent manner, a linear *scale-space representation* is proposed in [24, 34]. The basic idea is to embed the original signal into an one-parameter family of gradually smoothed signals, in which fine scale details are successively suppressed. It can be shown that the Gaussian kernel and its derivatives are one of the possible smoothing kernels for such scale-space. The Gaussian; kernel is well-suited for defining a space-scale because of its linearity and spatial shift invariance, and the notion

that structures at coarse scales should be related to structures at finer scales in a well-behaved manner (new structures are not created by the smoothing method). Scale-space representation is a special type of multiscale representation that comprises a continuous scale parameter and preserves the same spatial sampling at all scales. Formally, the linear-space representation of a continuous signal is constructed as follows. Let $f : \Re^N \rightarrow \Re$ represent any given signal. Then, the scale-space representation $L : \Re^N \times \Re_+ \rightarrow \Re$ is defined by $L(\cdot; 0) = f$ so that

$$L(\cdot; t) = g(\cdot; t) * f$$

where $t \in \Re_+$ is the scale parameter, and $g : \Re^N \times \Re_+ \setminus \{0\} \rightarrow \Re$ is the Gaussian kernel. In arbitrary dimensions, it is written as:

$$g(x; t) = \frac{1}{(2\pi t)^{N/2}} e^{-x^T x / (2t)} = \frac{1}{(2\pi t)^{N/2}} e^{-\sum_{i=1}^N x_i^2 / (2t)} \quad x \in \Re^N, x_i \in \Re$$

The square root of the scale parameter, $\sigma = \sqrt{t}$, is the standard deviation of the kernel g and is a natural measure of spatial scale in the smoothed signal at scale t . From this scale-space representation, multiscale spatial derivatives can be defined by

$$L_{x^n}(\cdot; t) = \partial_{x^n} L(\cdot; t) = g_{x^n}(\cdot; t) * f,$$

where g_{x^n} denotes a derivative of some order n .

The main idea behind the construction of this scale-space representation is that the fine scale information should be suppressed with increasing values of the scale parameter. Intuitively, when convolving a signal by a Gaussian kernel with standard deviation $\sigma = \sqrt{t}$, the effect of this operation is to suppress most of the structures in the signal with a characteristic length less than σ . Different directional derivatives can be used to extract different kind of structural features at different scales. It is shown in the literature [35] that a possible complete set of directional derivatives up to the third order are $\partial^n = [\partial_0, \partial_{90}, \partial_0^2, \partial_{60}^2, \partial_{120}^2, \partial_0^3, \partial_{45}^3, \partial_{90}^3, \partial_{135}^3]$. So our feature vector will consist on the directional derivatives, including the zero derivative, for each of the n scales desired:

$$F = \{\{\partial^n, G^n\}, n \in \Re\}$$

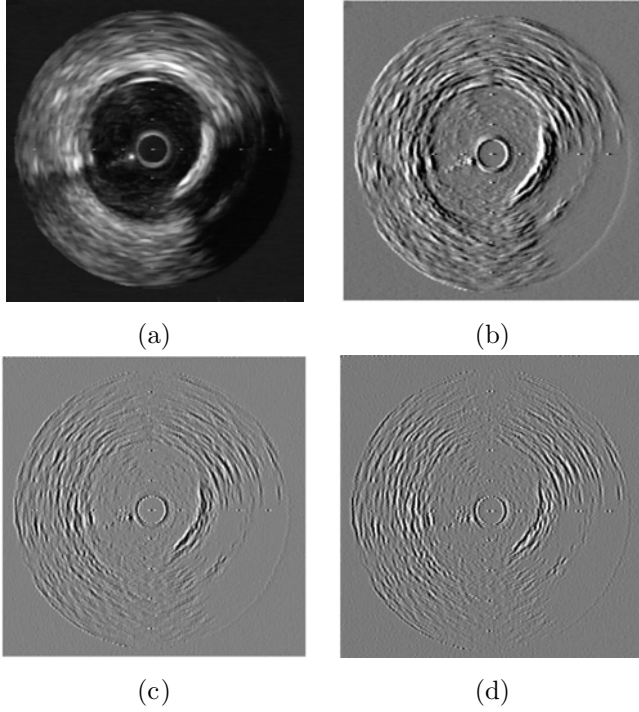


Figure 2.8: Derivative of Gaussian responses for $\sigma = 2$. (a) Original image; (b) first derivative of Gaussian response; (c) second derivative of Gaussian response; (d) third derivative of Gaussian response.

Figure 2.8 shows some of the responses for the derivative of Gaussian bank of filters for $\sigma = 2$. Figures 2.8(b), 2.8(c), and 2.8(d) display the first, second, and third derivatives of Gaussian, respectively.

2.2.2.2 Wavelets

Wavelets come to light as a tool to study nonstationary problems [36]. Wavelets perform a decomposition of a function as a sum of local bases with finite support and localized at different scales. Wavelets are characterized for being bounded functions with zero average. This implies that the shapes of these functions are waves restricted in time. Their time-frequency limitation yields a good location. So a wavelet ψ is a function of zero average:

$$\int_{-\infty}^{+\infty} \psi(t) dt = 0$$

which is dilated with a scale parameter s and translated by u :

$$\varphi_{u,s}(t) = \frac{1}{\sqrt{s}} \psi \left(\frac{t-u}{s} \right)$$

The wavelet transform of f at scale s and position u is computed by correlating f with a wavelet atom:

$$W_f(u, s) = \int_{-\infty}^{+\infty} f(t) \frac{1}{\sqrt{s}} \psi^* \left(\frac{t-u}{s} \right) dt \quad (2.1)$$

The continuous wavelet transform $W_f(u, s)$ is a two-dimensional representation of a one-dimensional signal f . This indicates the existence of some redundancy that can be reduced and even removed by subsampling the parameters of these transforms. Completely eliminating the redundancy is equivalent to building a basis of the signal space.

The decomposition of a signal gives a series of coefficients representing the signal in terms of the base from a mother wavelet, that is, the projection of the signal on the space formed by the base functions.

The continuous wavelet transform has two major drawbacks: the first, stated formerly, is redundancy and the second, impossibility to calculate it unless a discrete version is used. A way to discretize the dilation parameter is $a = a_0^m$, $m \in \mathbb{Z}$, $a_0 \neq 1$ constant. Thus, we get a series of wavelets ψ_m of width, a_0^m . Usually, we take $a_0 > 1$, although it is not important because m can be positive or negative. Often, a value of $a_0 = 2$ is taken. For $m = 0$, we make s to be the only integer multiples of a new constant s_0 . This constant is chosen in such a way that the translations of the mother wavelet, $\psi(t - ns_0)$, are as close as possible in order to cover the whole real line. Then, the election of s level is as follows:

$$\psi_{m,n}(t) = a_0^{-m/2} \psi \left(\frac{t - ns_0 a_0^m}{a_0^m} \right) = a_0^{-m/2} \psi(a_0^{-m} t - ns_0)$$

that covers the entire real axis as well as the translations $\psi(t - ns_0)$ does. Summarizing, the discrete wavelet transform consists of two discretizations in the transformation Eq. (2.1),

$$a = a_0^m, \quad b = nb_0 a_0^m, \quad m, n \in \mathbb{Z}, \quad a_0 > 1, b_0 > 0$$

The multiresolution analysis (MRA) tries to build orthonormal bases for a dyadic grid, where $a_0 = 2$, $b_0 = 1$, which besides have a compact support region. Finally, we can imagine the coefficients $d_{m,n}$ of the discrete wavelet transform as the sampling of the convolution of signal $f(t)$ with different filters $\psi_m(-t)$,

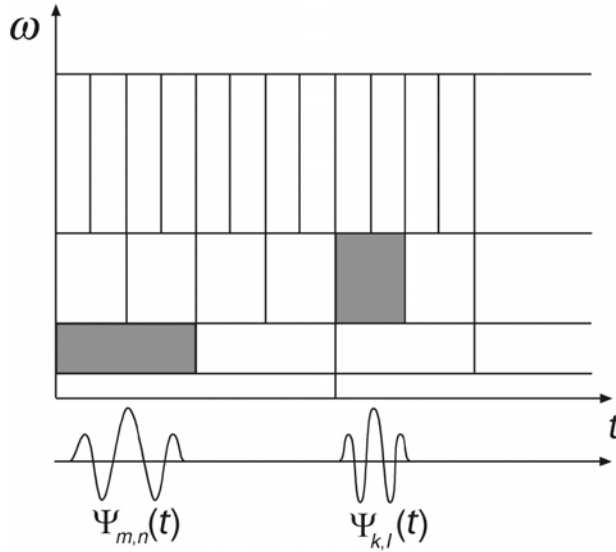


Figure 2.9: Scale-frequency domain of wavelets.

where $\psi_m(t) = a_0^{-m/2} \psi(a^{-m}t)$

$$y_m(t) = \int f(s) \psi_m(s - t) ds \quad d_{m,n} = y_m(na_0^m)$$

Figure 2.9 shows the dual effect of shrinking of the mother wavelet as the frequency increases, and the translation value decreasing as the frequency increases. The mother wavelet keeps its shape but if high-frequency analysis is desired the spatial support of the wavelet has to decrease. On the other hand, if the whole real line has to be covered by translations of the mother wavelet, as the spatial support of the wavelet decreases, the number of translations needed to cover the real line increases. Unlike Fourier transform, where translations of analysis are at the same distance for all the frequencies.

The choice of a representation of the wavelet transform leads us to define the concept of a *frame*. A frame is a complete set of functions that, though able to span $L^2(\mathfrak{R})$, is not a base because it lacks the property of linear independence. MRA proposed in [26] is another representation in which the signal is decomposed in an approximation at a certain level L with L detail terms of higher resolutions. The representation is an orthonormal decomposition instead of a redundant frame, and therefore, the number of samples that defines a signal is the same as that the number of coefficients of their transform. A MRA consists

of a sequence of function subspaces of successive approximation. Let P_j be an operator defined as the orthonormal projection of functions of L^2 over the space V_j . The projection of a function f over V_j is a new function that can be expressed as a linear combination of the functions that form the orthonormal base of V_j . Coefficients of the combination of each base function is the scalar product of f with the base functions:

$$P_j f = \sum_{n \in \mathbb{Z}} \langle f, \phi_{j,n} \rangle \phi_{j,n}$$

where

$$\langle f, g \rangle = \int_{-\infty}^{+\infty} f(t)g(t) dt$$

Earlier we have pointed out the nesting condition of the V_j spaces, $V_j \subset V_{j-1}$. Now, if $f \in V_{j-1}$ then $f \in V_j$ or f is orthonormal to all the V_j functions, that is, we divide V_{j-1} in two disjoint parts: V_j and other space W_j , such that if $f \in V_j, g \in W_j, f \perp g$; W_j is the orthonormal complement of V_j in V_{j-1} :

$$V_{j-1} = V_j \oplus W_j$$

where symbol \oplus measures the addition of orthonormal spaces. Applying the former equation and the completeness condition, then

$$\cdots \oplus W_{j-2} \oplus W_{j-1} \oplus W_j \oplus W_{j+1} \oplus \cdots = \bigoplus_{j \in \mathbb{Z}} W_j = L^2$$

So, we can write

$$P_{j-1} f = P_j f + \sum_{n \in \mathbb{Z}} \langle f, \psi_{j,n} \rangle \psi_{j,n}$$

From these equations some conclusions can be extracted. First, the projection of a signal f in a space V_j gives a new signal $P_j f$, an approximation of the initial signal. Secondly, we have a hierarchy of spaces, then $P_{j-1} f$ will be a better *approximation* (more reliable) than $P_j f$. Since V_{j-1} can be divided in two subspaces V_j and W_j , if V_j is an approximation space then W_j , which is the complementary orthonormal space, it is the *detail* space. The less the j , the finer the details.

$$\begin{aligned} V_j &= V_{j+1} \oplus W_{j+1} = V_{j+2} \oplus W_{j+2} = \cdots \\ &= V_L \oplus W_L \oplus W_{L-1} \oplus \cdots \oplus W_{j+1} \end{aligned}$$

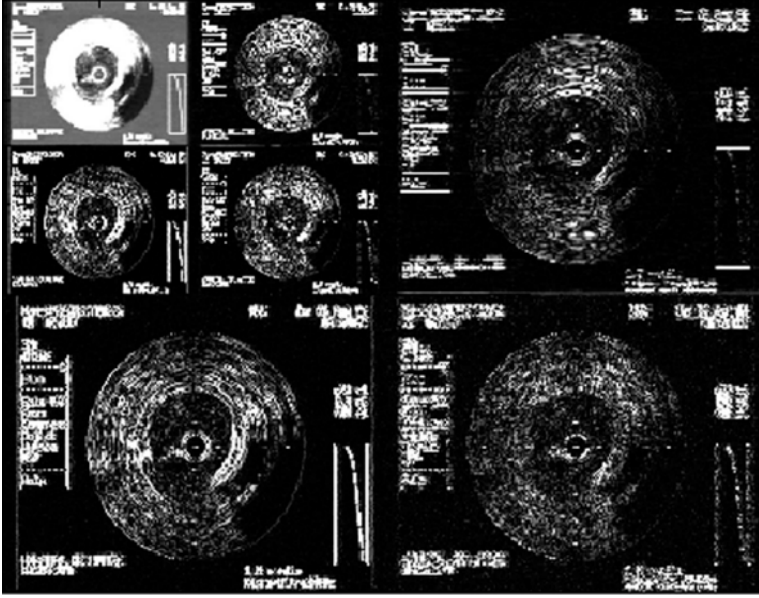


Figure 2.10: Wavelets multiresolution decomposition.

This can be viewed as a decomposition tree (see Fig. 2.10). At the top-left side of the image the approximation can be seen, and surrounding it the successive details. The further the detail is located the finer the information provided. So, the details at the bottom and at the right side of the image have information about the finer details and the smallest structures of the image decomposed. Therefore, we have a feature vector composed by the different detail approaches and the approximation for each of the pixels.

2.2.2.3 Gabor Filters

Gabor filters represent another multiresolution technique that relies on scale and direction of the contours [25,37]. The Gabor filter consists of a two-dimensional sinusoidal plane wave of a certain orientation and frequency that is modulated in amplitude by a two-dimensional Gaussian envelope. The spatial representation of the Gabor filter is as follows:

$$h(x, y) = \exp \left\{ -\frac{1}{2} \left[\frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2} \right] \right\} \cos(2\pi u_0 x + \phi)$$

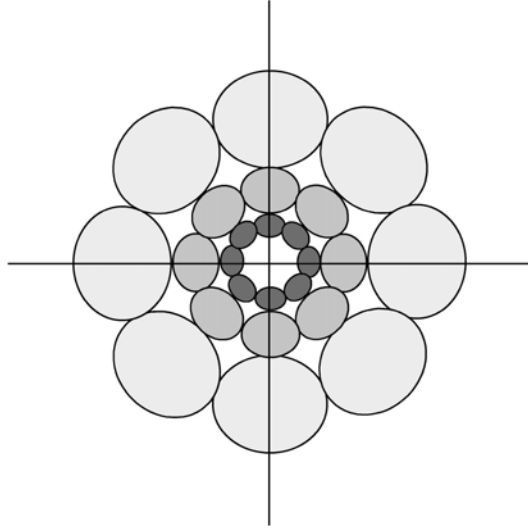


Figure 2.11: The filter set in the spatial-frequency domain.

where u_0 and ϕ are the frequency and phase of the sinusoidal plane wave along the x axis and σ_x and σ_y are the space constants of the Gaussian envelope along the x and y axis, respectively. Filters at different orientations can be created by rigid rotation of x - y coordinate system.

An interesting property of this kind of filters is their frequency and orientation-selection. This fact is better displayed in the frequency domain. Figure 2.11 shows the filter area in the frequency domain. We can observe that each of the filters has a certain domain defined by each of the leaves of the Gabor “rose.” Thus, each filter responds to a certain orientation and at a certain detail level. Wider the range of orientations, smaller the space filter dimensions and smaller the details captured by the filter, as bandwidth in the frequency domain is inversely related to filter scope in the space domain. Therefore, Gabor filters provide a trade-off between localization or resolution in both the spatial and the spatial-frequency domains. As it has been mentioned, different filters emerge from rotating the x - y coordinate system. For practical approaches one can use four angles $\theta_0 = 0^\circ, 45^\circ, 90^\circ, 135^\circ$. For an image array of N pixels (with N power of 2), the following values of u_0 are suggested [25,37]:

$$1\sqrt{2}, 2\sqrt{2}, 3\sqrt{2}, \dots, \text{ and } (N_c/4)\sqrt{2}$$

cycles per image width. Therefore, the orientations and bandwidth of such filters vary with 45° and 1 octave. These parameters are chosen because there is physiologic evidences of frequency bandwidth of simple cells in visual cortex being of about 1 octave, and Gabor filters try to mimic part of the human perceptual system.

The Gabor function is an approximation to a wavelet. However, though admissible, it does not result in an orthogonal decomposition, and therefore, a transformation based on Gabor's filters is redundant. On the other hand, Gabor filtering is designed to be nearly orthogonal, reducing the amount of overlap between filters.

Figure 2.12 shows different responses for different filters of the spectrum. Figures 2.12(a) and 2.12(b) correspond to the inner filters with reduced frequency bandwidth displayed in Fig. 2.11. It can be seen that they deliver only

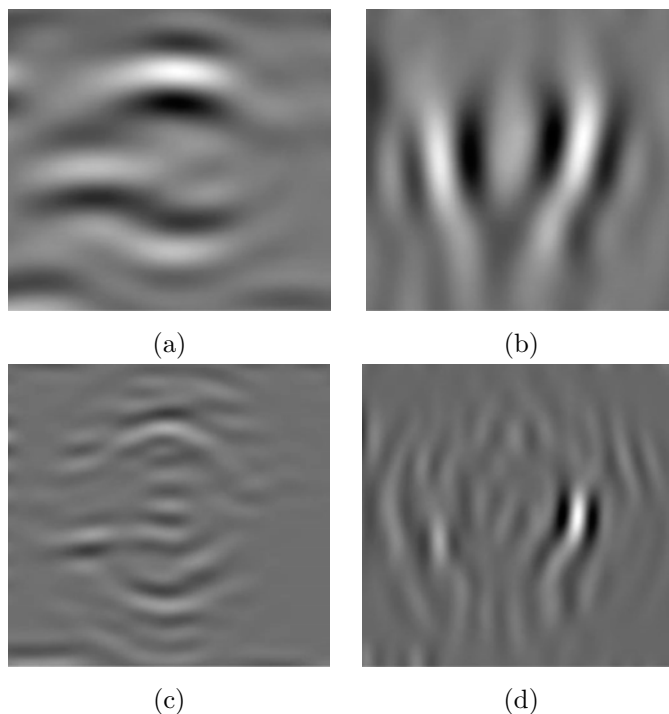


Figure 2.12: Gabor filter bank example responses. (a) Gabor vertical energy of a coarse filter response. (b) Gabor horizontal energy of a coarse filter response. (c) Gabor vertical energy of a detail filter response. (d) Gabor horizontal energy of a detail filter response.

Table 2.1: Dimensionality of the feature space provided by the texture feature extraction process

Method	Space dimension
Co-occurrence matrix measures	48
Accumulation local moments	81
Fractal Dimension	1
Local Binary Patterns	3
Derivative of Gaussian	60
Wavelets	31
Gabor's filters	20

coarse information of the structure and the borders are far from the original location. In the same way, Figs. 12(c) and 12(d) are filters located on a further ring, and therefore respond to details in the image.

It can be observed that the feature extraction process is a transformation of the original two-dimensional image domain to a feature space that probably will have different dimensions. In some cases, the feature space remains low, as in fractal dimension and local binary patterns, that with very few features try to describe the texture present in the image. However, several feature spaces require higher dimensions, such as accumulation local moments, co-occurrence matrix measures, or derivatives of Gaussian. Table 2.1 shows the dimensionality of the different spaces generated by the feature extraction process in our texture-based IVUS analysis.

The next step after the feature extraction is the classification process. As a result of the disparity of the dimensionality of the feature spaces, we have to choose a classification scheme that is able to deal with high dimensionality feature data.

2.3 Classification Process

Once completed the feature extraction process, we have a set of features disposed in feature vectors. Each feature vector is composed of all the feature measures computed at each pixel. Therefore, for each pixel we have an n -dimensional point in the feature space, where n is the number of features. This set of data is the input to the classification process. The classification process

is divided in two main categories: supervised and unsupervised learning. While supervised learning is based on a set of examples of each class that trains the classification process, the unsupervised learning is based on the geometry position of the data in the feature space and its possibility to be grouped in clusters.

In this chapter we are mainly concerned with supervised learning and classification, since we know exactly what classes we are seeking. Supervised classification techniques are usually divided in parametric and nonparametric. Parametric techniques rely on knowledge of the probability density function of each class. On the contrary, nonparametric classification, does not need the probability density function and is based on the geometrical arrangement of the points in the input space. We begin describing a nonparametric technique, *k-nearest neighbors*, that will serve as a ground truth to verify the discriminability of the different feature spaces. Since nonparametric techniques have high computational cost, we make some assumptions that lead to describe *maximum likelihood* classification techniques. However, the last techniques are very sensitive to the input space dimension. It has been shown in the former section that some feature spaces cast the two-dimensional image data to high-dimensional spaces. In order to deal with high-dimensional data, a dimensionality reduction is needed. The dimensionality reduction techniques are useful to create a meaningful set of data because the feature space is usually large in comparison to the number of samples retrieved. The most known technique for dimensionality reduction is *principal component analysis* (PCA) [38]. However, PCA is susceptible to errors depending on the arrangement of the data points in the training space, because it does not consider the different distributions of data clusters. In order to solve the deficiency of PCA in discrimination matters, *Fisher linear discriminant analysis* is introduced [38, 39]. In order to try to improve the classification rate of simple classifiers, combination of classifiers is proposed. One of the most important classification assembling process is *boosting*. The last part of this section is devoted to a particular class of boosting techniques, *Adaptive Boosting (AdaBoost)* [40, 41].

2.3.1 k-Nearest Neighbors

Voting k-nearest neighbors classification procedure is a very popular classification scheme that does not rely on any assumption concerning the structure of the underlying density function.

As any nonparametric technique, the resulting classification error is the smallest achievable error given a set of data. This is true because this technique implicitly estimates the density function of the data, and therefore, the classifier becomes the *Bayes classifier* if the density estimates converge to the true densities when an infinite number of samples are used [38].

In order to classify a test sample X , the k -nearest neighbors to the test sample are selected from the overall training data, and the number of neighbors from each class ω_i among the k selected samples is counted. The test sample is then classified to the class represented by a majority of the k -nearest neighbors. That is

$$\mathbf{k}_j = \max\{\mathbf{k}_1 \cdots \mathbf{k}_L\} \rightarrow X \in \omega_j$$

$$\mathbf{k}_1 + \cdots + \mathbf{k}_L = k$$

where \mathbf{k}_j is the number of neighbors from class ω_j , ($j = 1, \dots, L$) among the selected neighbors. Usually, the same metric is used to measure the distance to samples of each class.

Figure 2.13 shows an example of a 5-nearest neighbors process. Sample X will be classified as member of the light gray class since there are 3-nearest neighbors of the black class while there are only 2 members of the white class.

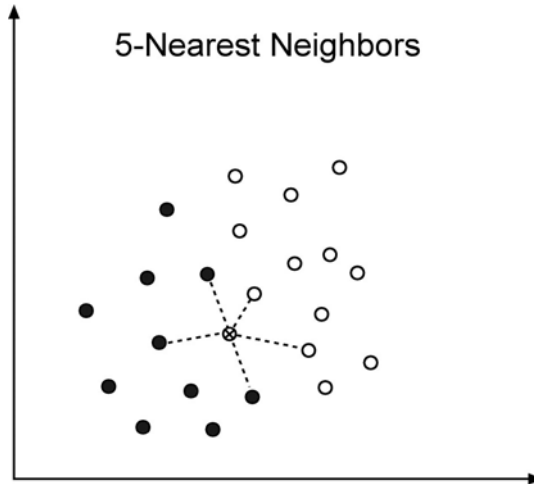


Figure 2.13: A 5-nearest neighbors example.

2.3.2 Maximum Likelihood

The maximum likelihood (ML) classifier is one of the most popular methods of classification [42]. The goal is to assign the most likely class w_j , from a set of N classes w_1, \dots, w_N , to each feature vector. The most likely class w_j from a given feature vector \mathbf{x} is the one with maximum posterior probability of belonging to the class $P(w_j | \mathbf{x})$. Using the Bayes' theorem, we have

$$P(w_j | \mathbf{x}) = \frac{P(\mathbf{x} | w_j)P(w_j)}{P(\mathbf{x})}$$

On the left side of the equation, there is the *a posteriori* probability of a feature vector \mathbf{x} to belong to the class w_j . On the right side, the *a priori* probability $P(\mathbf{x} | w_j)$ that expresses the probability of the feature vector \mathbf{x} being generated by the probability density function of w_j . $P(\mathbf{x})$ and $P(w_j)$ are the *a priori* probability of appearance of feature vector \mathbf{x} and the probability of appearance of each class w_j , respectively.

This model relies on the knowledge of the probability density function underlying each of the classes, as well as the probability of occurrence of the data and the classes. In order to reduce the complexity of such estimations, some assumptions are made. The first assumption generally made is the equiprobability of appearance for each of the feature vector as well as for each of the classes. This assumption reduces the Bayes' theorem to estimate the probability density function for each class:

$$P(w_j | \mathbf{x}) = P(\mathbf{x} | w_j)$$

Multiple methods can be used to estimate the *a priori* probability. Two of the most widespread methods are the assumption of a certain behavior and the mixture models.

A very common hypothesis is to identify the underlying probability density function with a multivariate normal distribution. In that case the likelihood value is

$$P(\mathbf{x} | w_j) = \frac{1}{\sqrt{\Sigma_j}(2\pi)^{n/2}} e^{-\frac{1}{2}(\mathbf{x}-\mu_j)\Sigma_j^{-1}(\mathbf{x}-\mu_j)^T}$$

where Σ_j and μ_j are the covariance matrix and the mean value for class j , respectively. In the case where the determinants of the covariance matrix for each of the classes are equal to each other, the likelihood value becomes the

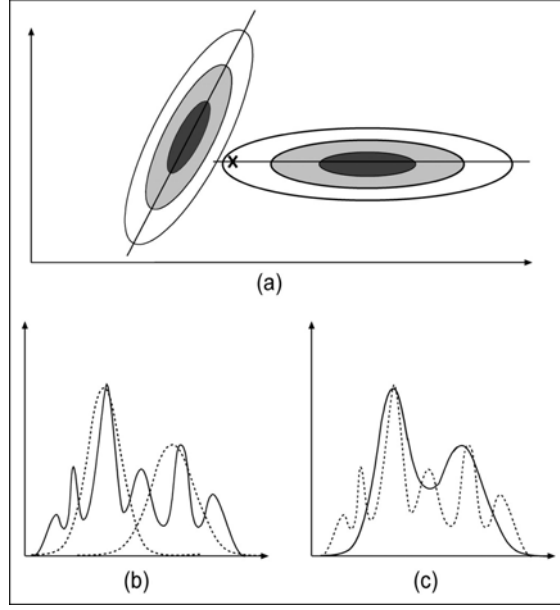


Figure 2.14: (a) Graphic example of the maximum likelihood classification assuming an underlying density model. (b) Unknown probability density function estimation by means of a 2 Gaussian mixture model. (c) Resulting approximation of the unknown density function.

same as the Mahalanobis distances. Figure 2.14(a) shows an example of the effect of this kind of classifier on a sample “X.” Although the sample seems to be nearer the left-hand distribution in terms of Euclidean distance, it is assigned to the class on the right hand since the probability of generating the sample is higher than its counterpart.

The other approach is to estimate the model of the probability density function. In the mixture model approach, we assume that the probability density function can be modelled using an ensemble of simple known distributions. If the base distribution is the Gaussian function it is called Gaussian mixture model. The interest in this method consists of the estimation of complex density function using low-level statistics.

The mixture model is composed of a sum of fundamental distributions, following the next expression:

$$p_i(x | \Theta) = \sum_{k=1}^C p_k(x | \theta_k) P_k \quad (2.2)$$

where C is the number of mixture components, P_k is the *a priori* probability of the component k , and θ_k represents the unknown mixture parameters. In our case, we have chosen Gaussian mixture models $\theta_k = \{P_k, \mu_k, \sigma_k\}$ for each set of texture data we want to model. Figures 2.14(b) and 2.14(c) show an approximation of a probability density function with a mixture of two Gaussian and the resulting approximation. Figure 2.14(b) shows the function to be estimated as a continuous line and the Gaussian functions used for the approximation as a dotted line. Figure 2.14(c) shows the resulting approximated function as a continuous line and the function to be estimated as a dotted line as a reference. One can observe that with a determined mixture of Gaussian distributions, an unknown probability density function can be well approximated. The main problem of this kind of approaches resides in its computational cost and the unknown number of base functions needed, as well as the value of their governing parameters. In order to estimate the parameters of each base distribution, general maximization methods are used, such as expectation-maximization (EM) algorithm [42].

However, this kind of techniques are not very suitable as the number of dimensions is large and the training data samples size is small. Therefore, a process of dimensionality reduction is needed to achieve a set of meaningful data. Principal component analysis and Fisher linear discriminant analysis are the most popular dimensionality reduction techniques used in the literature.

2.3.3 Feature Data Dimensionality Reduction

2.3.3.1 Principal Component Analysis

This method is also known as *Karhunen–Loeve* method [38]. Component analysis seeks directions or axes in the feature space that provide an improved, lower dimensional representation of the full data space. The method chooses a dimensionality reducing linear projection that maximizes the scatter of all projected samples. Let us consider a set of M samples $\{x_1, x_2, \dots, x_M\}$ in an n -dimensional space. We also consider a linear transformation that maps the original space in a lower dimensional space (of dimension m , $m < n$). The new feature vectors y are defined in the following way:

$$y_k = W^T x_k, \quad k = 1, \dots, M$$

where W is a matrix with orthonormal columns. The total scatter matrix S_T is defined as

$$S_T = \sum_{k=1}^M (x_k - \mu)(x_k - \mu)^T$$

where M is the number of samples and μ is the mean vector of all samples. Applying the linear transformation W^T , the scatter of the transformed feature vectors is $W^T S_T W$. PCA is defined as to maximize the determinant of the scatter of the transformed feature vectors:

$$W_{\text{opt}} = \text{argmax} |W^T S_T W| = [w_1 w_2 \cdots w_m]$$

where $\{w_i \mid i = 1, 2, \dots, m\}$ is the set of n -dimensional eigenvectors of S_T corresponding to the m largest eigenvalues.

Therefore, PCA seeks the directions of maximum scatter of the input data, which correspond to the eigenvectors of the covariance matrix having the largest eigenvalues. The n -dimensional mean vector μ and the $n \times n$ covariance matrix Σ are computed for the full dataset.

In summary, the eigenvectors and eigenvalues are computed and sorted in decreasing order. The k eigenvectors having the largest eigenvalues are chosen. With those vectors a $n \times m$ matrix W_{opt} is built. This transformation matrix defines an m -dimensional subspace. Therefore, the representation of the data onto this m -dimensional space is

$$y = A^t(x - \mu)$$

PCA is a general method to find the directions of maximum scatter of the set of samples. This fact however does not ensure that such directions will be optimal for classification. In fact, it is well known that some specific distributions of the samples of the classes result in projection directions that deteriorate the discriminability of the data. This effect is shown in Fig. 2.15 in which the loss of information when projecting to the PCA direction clearly hinders the discrimination process. Note that both projections of the clusters on the PCA subspace overlap.

2.3.3.2 Fisher Linear Discriminant Analysis

A classical approach to find a linear transformation that discriminates the clusters in an optimal way is discriminant analysis. Fisher linear discriminant

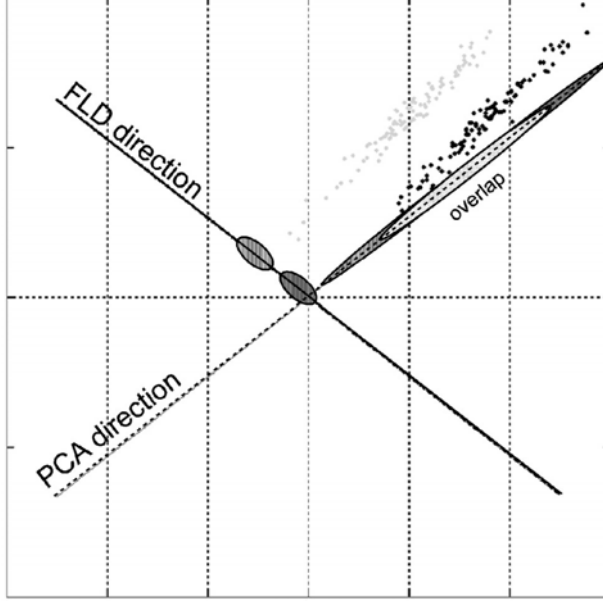


Figure 2.15: Example of the resulting direction using principal component analysis (PCA) and Fisher linear discriminant (FLD).

analysis [38, 39] seeks a transformation matrix W such that the ratio of the between-class scatter and the within-class scatter is maximized. Let the between-class scatter S_B be defined as follows:

$$S_B = \sum_{i=1}^c N_i (\mu_i - \mu)(\mu_i - \mu)^T \quad (2.3)$$

where μ_i is the mean value of class X_i , μ is the mean value of the whole data, c is the number of classes, and N_i is the number of samples in class X_i . Let the within-class scatter be

$$S_W = \sum_{i=1}^c \sum_{x_{k,i} \in X_i} (x_{k,i} - \mu_i)(x_{k,i} - \mu_i)^T \quad (2.4)$$

where μ_i is the mean value of class X_i , c is the number of classes, and N_i is the number of samples in class X_i . If S_W is not singular, the optimal projection matrix W_{opt} is chosen as the matrix that maximizes the ratio of the determinant of the between-class scatter matrix of the projected samples to the determinant

of the within-class scatter matrix of the projected samples:

$$W_{\text{opt}} = \operatorname{argmax}_W \frac{|W^T S_B W|}{|W^T S_W W|} = [\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_m] \quad (2.5)$$

where $\mathbf{w}_i, i = 1 \dots m$, is the set of S_W -generalized eigenvectors of S_B corresponding to the m largest generalized eigenvalues.

Opposed to PCA behavior, Fisher linear discriminant (FLD) emphasizes the direction in which both classes can be better discriminated. FLD uses more information about the problem as the number of classes and the samples in each of the classes must be known *a priori*. In Fig. 2.15 the projections on the FLD subspace are well separated.

In real problems, it can occur that it is not possible to find an optimal classifier. A solution is presented by assembling different classifiers.

2.3.4 AdaBoost Procedure

Adaptative Boosting (AdaBoost) is an arcing method that allows the designer to continue adding “weak” classifiers until some desired low-training error has been achieved [40, 41]. A weight is assigned to each of the feature points, these weights measure how accurate the feature point is being classified. If it is accurately classified, then its probability of being used in subsequent learners is reduced or emphasized otherwise. This way, AdaBoost focuses on difficult training points.

Figure 2.16 shows a diagram of the general process of boosting. The input data is resampled according to the weights of each feature data. The higher the weight the most probable the feature point will be in the next classification. The new set of feature points are inputs of the new classifier to be added to the process. At the end of the process, the responses of all the classifiers are combined to form the “strong” classifier.

AdaBoost is capable of performing a feature selection process while training. In order to perform both tasks, feature selection and classification process, a weak learning algorithm is designed to select the single features that best separate the different classes. For each feature, the weak learner determines the optimal classification function, so that the minimum number of feature points is misclassified. The algorithm is described as follows:

- Determine a supervised set of feature points $\{x_i, c_i\}$ where $c_i = \{-1, 1\}$ is the class associated to each of the feature classes.

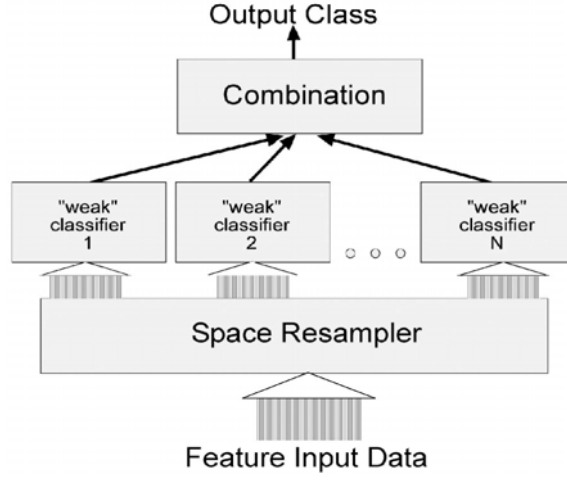


Figure 2.16: Block diagram of the AdaBoost procedure.

- Initialize weights $w_{1,i} = \frac{1}{2}m, \frac{1}{2l}$ for $c_i = \{-1, 1\}$ respectively, where m and l are the number of feature points for each class.
- For $t = 1..T$
 - Normalize weights

$$w_{t,i} \leftarrow \frac{w_{t,i}}{\sum_{j=1}^n w_{t,i}}$$

so that w_t is a probability distribution.

- For each feature, j train a classifier, h_j which is restricted to using a single feature. The error is evaluated with respect to w_t , $\epsilon_j = \sum_i w_i |h_j(x_i) - c_i|$.
- Choose the classifier, h_t with the lowest error ϵ_t .
- Update the weights:

$$w_{t+1,i} = w_{t,i} \beta_t^{e_i}$$

where $e_i = 1$ for each well-classified feature and $e_i = 0$ otherwise.

$\beta_t = \frac{\epsilon_t}{1-\epsilon_t}$. Calculate parameter $\alpha_t = -\log(\beta_t)$.

- The final strong classifier is

$$h(x) = \begin{cases} 1 & \sum_{t=1}^T \alpha_t h_t(x) \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

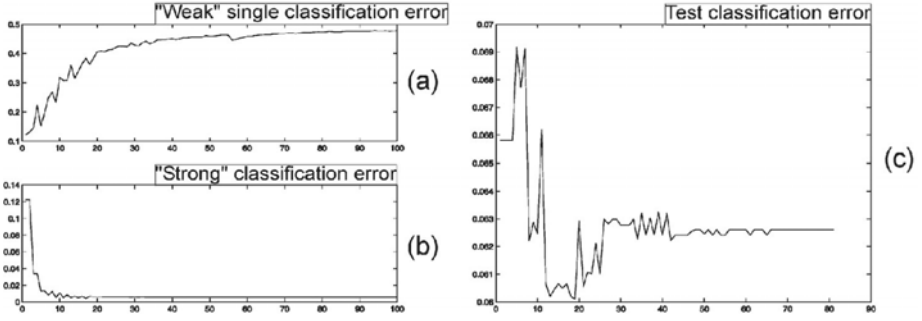


Figure 2.17: Error rates associated to the AdaBoost process. (a) Weak single classification error. (b) Strong classification error on the training data. (c) Test error rate.

Therefore, the strong classifier is the ensemble of a series of simple classifiers, $h_t(x)$, called “weak”. Parameter α_t is the weighting factor of each of the classifiers. The loop ends when the classification error of a weak classifier is over 0.5, the estimated error for the whole strong classifier is lower than a given error rate or if we achieve the desired number of weaks. The final classification is the result of the weighted classifications of the weaks. The process is designed so that if $h(x) > 0$, then pixel x belongs to one of the classes.

Figure 2.17 shows the evolution of the error rates for the training and the test feature points. Figure 2.17(a) shows the error evolution of each of the weak classifiers. The abscise axis is the number of the weak classifier, and the ordinate axis is the error percentage of a single weak. The figure illustrates how the error increases as more weak classifiers are added. This is because each new weak classifier focusses on the misclassified data of the overall system. Figure 2.17(b) shows the error rate of the system response on the training data. The abscise axis represents the number of iterations, that is, the number of classifiers added to the ensemble. As it is expected, the error rate decreases to very low values. This, however, does not ensure a test classification error of such accuracy. Figure 2.17(c) shows the test error rate. One can observe, that the overall error has a decreasing tendency as more weak classifiers are added to the process.

Therefore, the weak classifier has a very important role in the procedure. Different approaches can be used; however, it is relatively interesting to center our attention in low-time-consuming classifiers.

The first and the most straight forward approach to a weak is the perceptron. The perceptron is constituted by a weighed sum of the inputs and an adaptative

threshold function. This scheme is easy to embed in the adaboost process since it relies on the weights to make the classification.

Another approach to be taken in consideration is to model the feature points as Gaussian distributions. This allows us to define a simple scheme by simply calculating the weighed mean and weighed covariance of the classes at each step t of the process:

$$\mu_{i,t}^j = \sum_i w_{i,t} x_i \quad \Sigma_{i,t}^j = \sum_i w_{i,t} (x_i - \mu_{i,t}^j)^2$$

for each x_i^j point in class C_j . $W_{i,j}$ are the weights for each data point.

If feature selection is desired, this scheme is highly constrained to the N features of the N -dimensional feature space. If N is not enough large, the procedure could not improve its performance.

Both, the feature extraction and the classification processes, are the central parts of the tissue characterization framework. Next section is devoted to explain the different frameworks where these processes are applied for tissue characterization of IVUS images as well as provide quantitative results of their performance.

2.4 Results and Conclusions

The goal of automatic tissue characterization is to identify the different kind of plaques in IVUS images. This process requires two tasks: identification of what the plaque is and labelling of the different areas of the plaque. Figure 2.18 illustrates roughly the procedure of supervised tissue characterization. The IVUS image (see Fig. 2.18(a)) is preprocessed and sent to the automatic tissue characterization system. Figure 2.18(b) illustrates the physicians-assisted segmentation of IVUS, which will constitute a part of the training dataset. Figure 2.18(c) shows the first step, the accurate location of the lumen–plaque border and the adventitia border. Between both borders, the plaque is the region of interest to be classified. Figure 2.18(d) exemplifies the tissue characterization process. We are focussed on the plaque, and try to find and label areas corresponding to different plaques. In the figure, light gray areas are soft plaque, white areas are fibrotic plaque, and dark gray areas are calcium plaque. At the end of the process we obtain the tissue characterized IVUS (Fig. 2.18(e)) to be used by

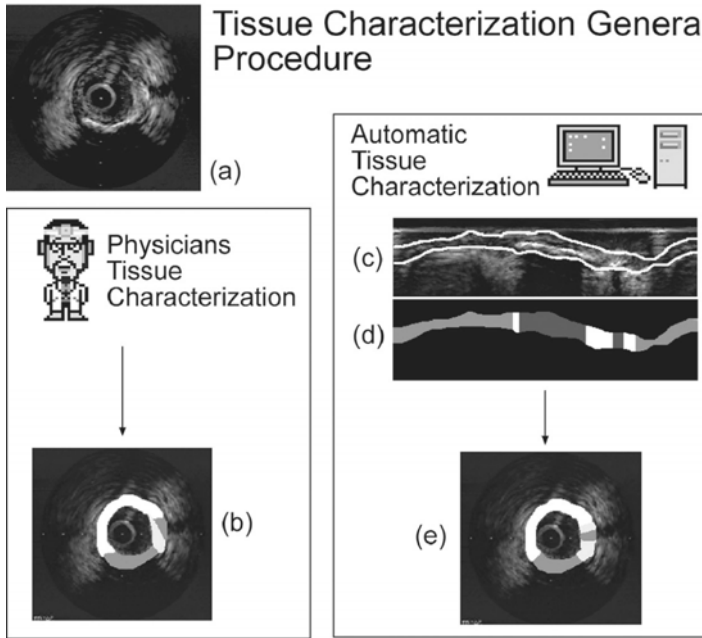


Figure 2.18: The tissue characterization process can be done manually by physicians (a) and (b) or by an automatic process (a), (c), (d), and (e). (c) Segmentation of the lumen-plaque border and adventitia border. (d) Processed image of the plaque characterization. (e) Final IVUS tissue characterization.

the physicians. These results have been used to validate against the manually segmented plaque regions (Fig. 2.18(b)).

Therefore, though we are concerned with tissue characterization, we cannot forget the segmentation of the plaque. A brief review on how to segment the plaque is exposed in the next section.

2.4.1 Segmentation of the Plaque

The segmentation of the plaque is a really important step before tissue characterization. There are multiple ways to achieve this goal [19, 20, 43–45]. In particular, we will focus on two general lines of work: the line proposed in [43] and the line proposed in [19, 20].

In [43] the process of segmentation relies on a manual definition of a region of interest. Using that region of interest a Sobel-like edge operator with

neighborhoods of 5×5 and 7×7 is applied. Once we have these features extracted, the problem of identification of the vessel wall and plaque border is solved by finding the optimal path in a two-dimensional graph. The key of the graph searching is to find the appropriate cost functions. In the paper, the authors propose different cost functions depending on whether the lumen–plaque border or the adventitia border is desired.

Having in mind the tissue classification goal of the process, in [19, 20], the authors try to find a segmentation of the overall tissue independent of what kind of tissue it is, to distinguish the lumen–plaque border. Therefore, the method consists of selecting a feature space and a classifier. This method takes advantage of the fact that for tissue characterization the same scheme must be used. Thus, a classifier is trained for general tissue discrimination. Hence, in the overall process the feature extraction process is performed once for both, plaque segmentation and tissue identification. What is different in both approaches is the classification selection and training data, and the post processing steps.

The classification step is performed using a fast classifier, boosting methods, or ML. The result of this step is a series of unconnected areas that are related to tissues. In order to find the exact location of the lumen–plaque border, a fast parametric snake is let to deform over the unconnected areas [46]. The snake performs a double task: first, it finds a continuous boundary between blood and plaque. The second task is that it ensures an interpolation and a fill-in process in regions where tissue is not located or not reliable (such as areas with reverberations due to the guide-wire, etc). The adventitia border is found by context using a 5×5 Sobel-like operator and deformable models. Figure 2.19 shows an example of a possible scheme for border location. First the IVUS image is transformed to cartesian coordinates. Then, a texture feature extraction step is performed. A classification scheme is trained to distinguish blood from tissue. At the end, a snake deforms to adapt to the classified IVUS image and locates accurately the blood–plaque border.

2.4.2 Tissue Characterization

2.4.2.1 Methodology

We begin our process of tissue characterization taking the IVUS image and transforming it to cartesian coordinates (Fig. 2.20(a)). Once the cartesian

Methodology:

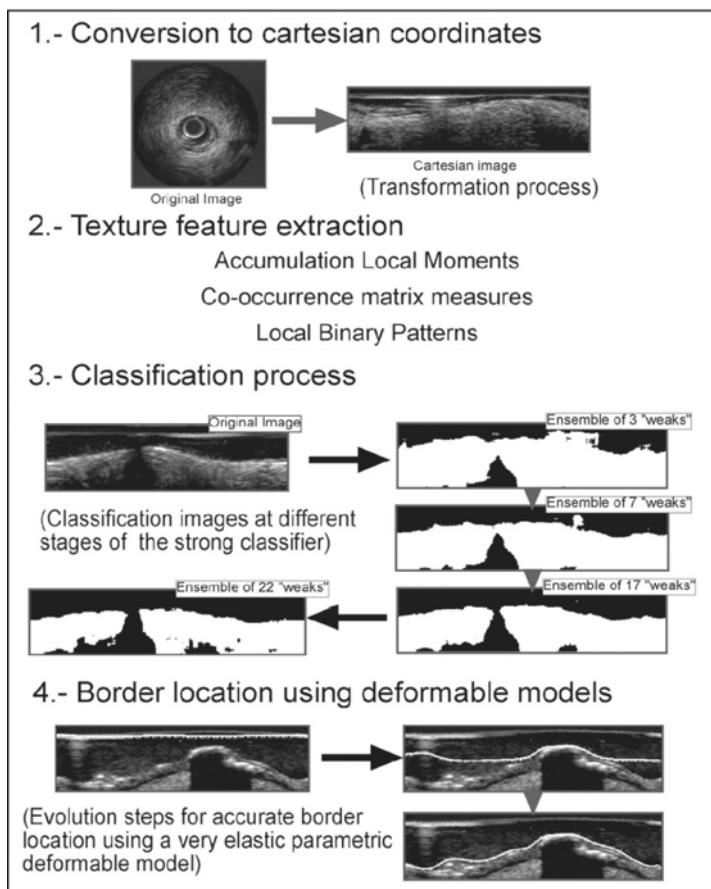


Figure 2.19: Plaque segmentation (see text).

transformation is done, artifacts are removed from the image (Fig. 2.20(b)). There are three main artifacts in an IVUS image: the transducer wrapping, which creates a first halo at the center of the image (in the cartesian image the echo is shown at the top of the image); the guide-wire effect, which produces an echo reverberation near the transducer wrapping; and the calibration grid, which are markers at a fixed location that allow the physicians to evaluate quantitatively the morphology and the lesions in the vessel. With the artifacts removed, we proceed to identify intima and adventitia using the process described in the former section. At this point, we have the plaque located and we are concerned with tissue identification (Fig. 2.20(c)). The tissue classification process is divided

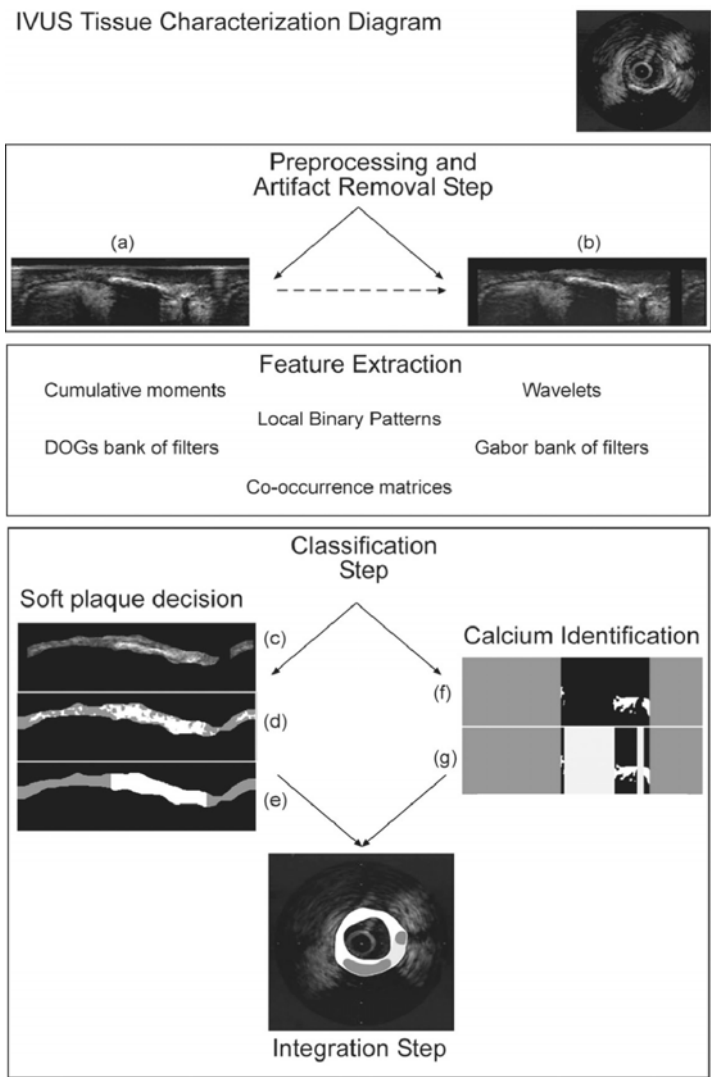


Figure 2.20: Tissue characterization diagram.

in three stages: First, the soft–hard classification (Figs. 2.20(d) and 2.20(e)), in which the soft plaque, the hard plaque, and calcium are separated. In the second stage, the calcium is separated from the hard plaque (Figs. 2.20(f) and 2.20(g)). At the last stage, the information is fused and the characterization is completed. We will refer later to this diagram to explain some parts of the process. Recall that the plaque is the area comprised between the intima and the adventitia. With both borders located we can focus on the tissue of that area.

For such task, the three stages scheme formerly described is used. Regarding the first stage of the process, a classification is performed on the feature space. At this point, a feature space and a classifier must be selected. To help to choose which feature space and which classifier to use, we try each of the feature spaces with a general purpose classifier, the k-nearest neighbors method used as a ground truth classifier. Regardless the classifier used, the information provided at the output of the system is a pixel classification. Using these data we can further process the classification result incorporating region information from the classification process and obtain clear and smoother borders of the soft and the mixed plaques. Different processes can be applied to achieve this goal, two possible approaches are region-based area filtering and classification by density filtering. In a region-based area filtering the less significant regions in terms of size are removed from the classification. On the other hand, the other method relies on keeping the regions that have high density of classification responses. As the classification exclusively aims to distinguish between soft and hard plaque, a separate process is added to separate hard plaque from calcium.

Once soft and hard plaque are distinguished, we proceed to identify what part of the hard plaque corresponds to calcium. One can argue why not to include a third class in the previous classifier. The reason we prefer not to do so is because experts' identification of calcium plaques is performed by context. Experts use the shadowing produced by the absorption of the echoes, behind a high-echoreflective area, to label a certain area as calcium. In the same way, we take the same approach. On the other hand, the fact of including a third class only hinders the decision process and increases the classifier complexity. Therefore, the calcium identification process is made by finding the shadowing areas behind hard plaque. Those areas are easily identified because the soft-hard classification also provides this information (Fig. 2.20) since shadowing areas are classified as nontissue. We can see a plausible way of finding calcified areas. Figure 2.20(f) shows the classification result under the adventitia border of the "hard" tissue. Dark gray level areas are regions with soft plaque and, therefore, do not provide information of the calcium composition of the plaque. We use one of the previous classified images, the soft-hard classification or the blood-plaque classification. In white, it is displayed the regions of tissue under the adventitia border in the area of interest. Figure 2.20(g) shows in light gray the areas of shadowing, and therefore, the areas labelled as calcium.

To end the process, the last stage is devoted to recast the resulting classification to its original polar domain by means of a simple coordinate transformation.

2.4.2.2 Experimental Results

To evaluate the results, a classification of over 200 full-tissue regions from 20 different patients has been performed. The training set is a subset of two thirds of the overall data determined using the bootstrapping strategy. The rest of the data has been used as test set. Previously, different physicians have determined and delineated plaque regions in each full-tissue image.

The first experiment is to set a ground truth for the feature spaces, as a measure to evaluate their description power. We have used k-nearest neighbors as a ground truth classifier. To choose the number of neighbors, we select a feature space and evaluate the performance for different values of k . Tables 2.2, 2.3 and 2.5 show the error rates for pixel classification (RAW Error) and postprocessed classification taking into account neighboring information and density of classifier cluster responses (Post Error). These tables also show the percentages of false positives (FP) and false negatives (FN) for both errors. The FP and FN are included as they give information of the possible geometry of samples in the feature space.

Table 2.2 illustrates the results regarding the selection of the number of neighbors k . It can be seen that for $k = 7$ a lower pixel error rate is obtained. Therefore, the performance of the feature spaces will be evaluated using 7-nearest neighbors. The result of the classification of the test data using all feature spaces and 7-nearest neighbors classifier is shown in Table 2.3. Observing the RAW data error rate, the best overall feature spaces are the co-occurrence matrices, local binary patterns, derivatives of Gaussian, and accumulation local moments. These results are confirmed looking at the postprocessing error rate and ratifies

Table 2.2: Selection of the parameter k , using local binary pattern feature space as a reference

k value	RAW error	FP	FN	Post error	FP	FN
3	33.94	25.13	8.80	10.16	3.46	6.69
7	25.67	9.67	16.23	3.45	2.67	0.81
15	32.93	26.19	6.74	5.81	3.46	2.34

Table 2.3: Feature space performance discriminating hard plaque from soft plaque using k-nearest neighbor

Feature space	RAW error	FP	FN	Post error	FP	FN
Co-occurrence measures	22.36	10.91	11.45	10.88	4.19	6.68
Derivative of Gaussian	27.81	23.51	4.95	16.29	16.67	0.04
Gabor filters	35.26	18.86	17.22	16.26	16.49	0.07
Wavelets	45.05	20.52	24.90	31.78	24.40	7.68
Accumulation local moments	31.72	16.42	15.30	12.17	11.36	0.81
Local binary patterns	25.67	9.67	16.23	3.45	2.67	0.81

the qualitative evaluation shown in Table 2.4, where we observe that the same feature spaces are the ones that perform best. Analyzing each of the feature spaces in terms of FP and FN rates, we can deduce that Co-occurrence feature space has good discrimination power, having a “symmetric” nature where both FP and FN rates are comparable. In the same sense, we can deduce that the overlapping of both classes is similar. Derivatives of Gaussian’s filter space have tendency to over-classify hard plaque. The classes in the feature space are not very well defined as hard plaque must have a higher scatter than the soft plaque. Gabor filter’s bank gives a good description of both classes as they have similar false rates. However, both classes are very overlapped giving a hard time to the classification process. Wavelets overlapping of classes in the feature space is extremely high; therefore, it describes bad each of the classes. Accumulation local moments have similar description power than Gabor filter’s bank; however, the different responses from both allow a much better postprocessing in accumulation local moments. This fact allows us to suppose that the classification error points in the image domain are much more scattered and

Table 2.4: Descriptive table of the discriminative power of each feature space using k-nearest neighbors

Feature space	Qualitative speed	Qualitative performance
Co-occurrence measures	Slow	Good
Gabor space	Slow	Acceptable
Wavelets	Fast	Poor
Derivative of Gaussian	Slow	Acceptable
Accumulation local moments	Fast	Good
Local binary patterns	Fast	Good

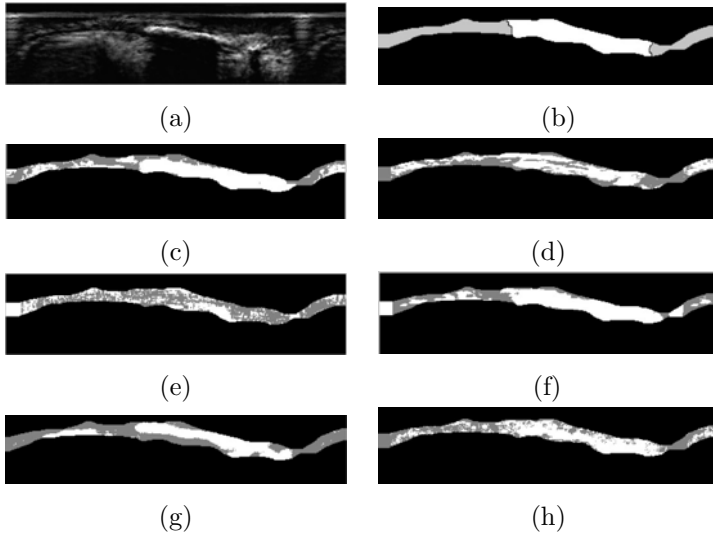


Figure 2.21: Tissue pixel classification data using 7-nearest neighbors method on different feature spaces. (a) Original image in cartesian coordinates. (b) Expert manual classification of tissue. (c) Co-occurrence feature space. (d) Gabor feature space. (e) Wavelets feature space. (f) Derivative of Gaussian feature space. (g) Accumulation local moments feature space. (h) Local binary patterns feature space.

with very few local density. Local binary patterns have good descriptive power as well as giving a more sparse pattern in FP and FN in the image domain. Figure 2.21 provides a graphical example of the performance of 7-nearest neighbors method applied to several feature spaces. Observing the images, we realize that scale-space processes, derivative of Gaussian, Gabor filters, and wavelets have poor to acceptable discrimination power, and therefore, are not suitable for the task of tissue discrimination. On the other hand, statistic-based feature spaces and structure feature spaces have acceptable to good performances. Table 2.4 details the conclusions arisen from the experiment. The qualitative speed nomenclature (fast/slow) indicates the viability of the feature space technique to be included in a real time or near-real time process. A “fast” scheme denotes a method over 10 times faster than the “slow” one. Because the results are obtained using prototypes and not a full application, no absolute time measure is provided. Note, also, that the images displayed are pixel-based classification results and have no further smoothing postprocessing. To further develop our

Table 2.5: Feature space performance using FLD and Mahalanobis distance

Feature space	RAW error	FP	FN	Post error	FP	FN
Co-occurrence measures	40.88	34.66	6.20	12.91	12.10	0.81
Accumulation local moments	35.50	20.34	16.16	13.83	13.02	0.81
Local binary patterns	26.37	5.76	20.85	6.93	1.52	5.47

discussion we will only take the three best postprocessed data performing feature spaces: co-occurrence matrix measures, accumulation local moments, and local binary patterns. Up to this point we have neither taken into account complexity of the methods nor time issues. However, these are critical parameters in real applications, thus, we consider them in our following discussions.

Once the feature space is selected, the next decision is to find the most suitable classifier taking into account our problem constraints, if any. We are concerned with speed issues, therefore, simple but powerful classifiers are required. Because the high dimensionality of two of the feature spaces selected (co-occurrence matrix measures have about 24 features per distance and accumulation local moments have 81 features) a dimensionality reduction step is desired. PCA is the first obvious choice, but because great amount of overlapping data the results are worse than using Fisher's linear discriminant analysis which is focalized in finding the most discriminative axes for our given set of data. The result of this experiment is shown in Table 2.5. We use maximum likelihood combined with a Fisher linear discriminant analysis reduction. As local binary patterns do not need dimensionality reduction due to the small amount of features computed (three features), the comparison with this method is done by just classifying with the ML method. As expected, the raw results are much worse with this kind of classifier. Co-occurrence matrix measures take the worse part doubling their error rate. However, local binary patterns, though they have also worse error rate with ML, manage to be the most discriminative of the three methods. This fact is also shown in the postprocessing, where local binary patterns still have the lower error ratio. Co-occurrence matrix measures regain their discrimination power after the postprocessing.

Therefore, using one of the fastest classifiers, ML, one achieves, at least, a classification ratio over 87% (with accumulation local moments). If the selected feature space is local binary patterns, the scheme is the fastest possible scheme as local binary patterns are computationally efficient and low-time consuming as

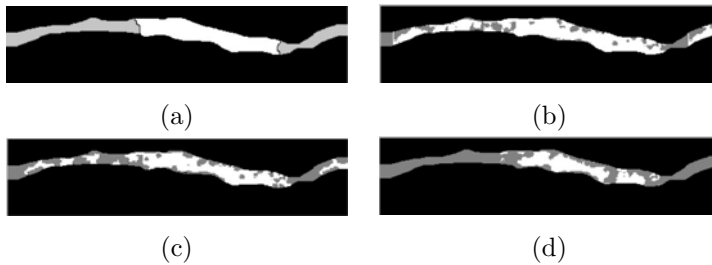


Figure 2.22: Boosting procedure for tissue characterization at different stages of its progress. (a) Expected hand classification by an expert. (b) First stage of the boosting procedure. (c) Classification with a five classifiers ensemble. (d) Classification with 10 “weak” ensemble.

well as the ML classifier does not transform data in another feature space. This scheme is really well suited for real-time or near-real-time applications because of both time efficiency and reliability in the classification. This is, however, by no means the only near-real-time configuration available since accumulation local moments are computationally as fast as local binary patterns. However, the FLD dimensionality reduction hinders the process due to the complexity of the data in its original feature space. To overcome this problem, other classifiers can be used. The necessity to find reliable and fast classifiers lead us to boosting techniques. Boosting techniques allow a fast and simple classification procedure to improve its performance as well as maintaining part of its speed. To illustrate this fact Fig. 2.22 shows the evolution of the classification when more classifiers are added to the *strong classifier*. Figure 2.22(a) shows the expected hand classification by a physician. Figure 2.22(b) shows the base classification of a single “weak”. Figure 2.22(c) illustrates the result of the classification using an ensemble of five classifiers. Figure 2.22(d) shows the resulting classification after the addition of 10 weak classifiers to the ensemble. The error rates at different stages of the process are also shown in Table 2.6. These results are computed using a ML method as a *weak classifier* on the accumulation local moments space. The numbers show how the error rate is improved, and, though the raw classification error rate is nearly immutable, we can observe that there is a great change in the classification data points distribution in the image domain since the FP and FN rates drastically change. The postprocessing error rate gives better description of what is happening. The disposition of the error points in the classification image is more sparse and unrelated to their neighborhood, allowing better

Table 2.6: Error rates using boosting methods with maximum likelihood with the accumulation local moments space

Ensemble no.	RAW error	FP	FN	Post error	FP	FN
Base error	34.86	28.20	6.98	41.94	40.33	1.10
Ensemble of 5 c.	29.38	16.32	13.32	33.17	31.87	1.10
Ensemble of 10 c.	31.44	7.36	23.37	7.92	3.22	4.76

postprocessing and classification rates. In this case, the classification rate is over 92% with a classifier as fast as applying 10 times a threshold. Therefore, using accumulation local moments and boosting techniques we have another fast and highly accurate scheme for real-time or near-real-time tissue characterization.

Up to this point, we have discussed the reliability of the soft plaque versus hard plaque discrimination process, which is our main concern, since the identification of calcium is reduced to the part of hard plaque with a large shadowing area. Using the method described in the former section, 99% of the calcium plaque is correctly identified. Figure 2.23 shows some results of the tissue characterization process. Figures 2.23(a) and 2.23(b) show the characterization of a soft plaque. In Figs. 2.23(c) and 2.23(d), there are two different kind of plaques detected, calcium (gray region) and soft plaque (white region). Figures 2.23(e) and 2.23(f) show the characterization of the three kind of plaques: fibrotic (light gray region), soft plaque (white region), and calcium (dark gray region).

2.4.3 Conclusions

Tissue characterization in IVUS images is a crucial problem for the physicians for studying the vascular diseases. However, this task is complex and suffers from multiple drawbacks (slow manual process, subjective interpretation, etc.) Therefore, automatic plaque characterization is a highly desirable tool.

However, automatic tissue characterization is a problem of high complexity. First of all, we need a unique and powerful description of the tissues to be classified. This is done by the feature extraction process, that in order to obtain complete and meaningful description, image features should be based on texture. Thus, a study of the most representable feature spaces is done, to conclude with some enlightening results. After analyzing the experimental results, we conclude that co-occurrence matrix measures, local binary patterns, and

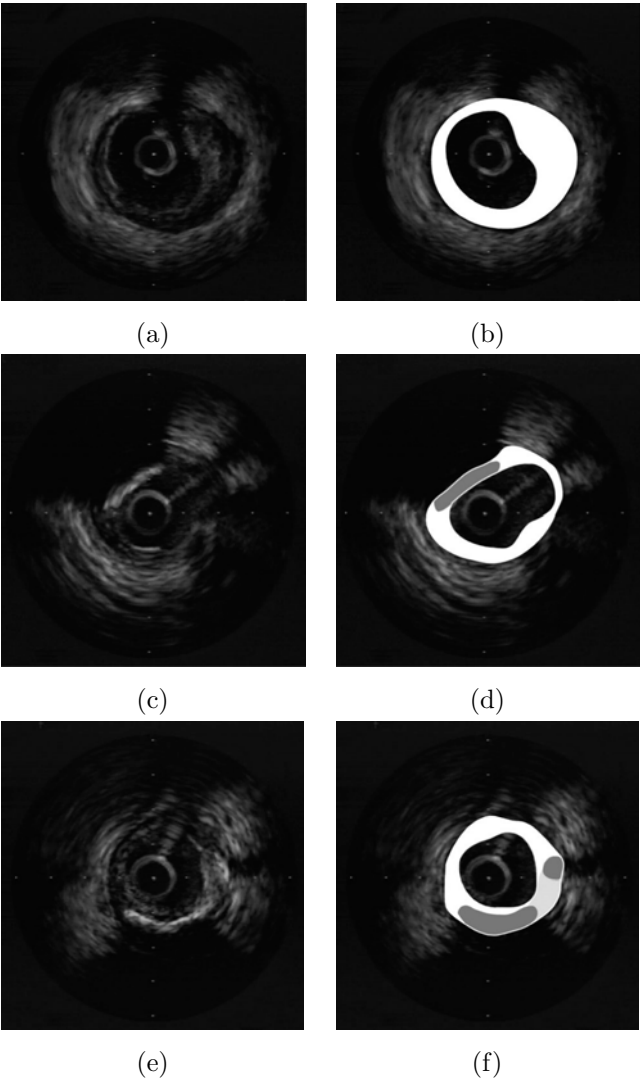


Figure 2.23: Tissue characterization results: (b), (d), and (f) White labels soft plaque, dark gray areas are displayed where calcium plaques are located, and light gray areas labels hard plaque. (a), (c), and (e) Original images.

accumulation local moments are good descriptors of the different kind of plaque tissues. However, local binary patterns and accumulation local moments are also fast, in terms of low-time processing. On the other hand, the classification of the feature data is a critical step. Different approaches to the classification problem are described and proposed as candidates in our framework. We proved that

k-nearest neighbor method gives the best performance as a classifier. But, ML and methods based on an ensemble of classifiers have high discrimination rate and lower classification time. Therefore, two real-time or near-real-time approaches are proposed: The first method combines local binary patterns with ML methods. The second method uses accumulation local moments and boosting techniques.

In conclusion, we have presented a general fully automatic and real-time or near-real-time framework with high accuracy plaque recognition rate for tissue characterization in IVUS images.

Questions

1. *What is tissue characterization in IVUS images?*
2. *Why is automatic tissue characterization an important issue?*
3. *Why do we use texture-based descriptors?*
4. *Why do we use supervised classification?*
5. *What is the feature space?*
6. *Why is dimensionality reduction needed in the classification process?*
7. *What is the main idea under the boosting classification?*
8. *What is the segmentation of the plaque for?*
9. *Discuss the methodology for the tissue characterization framework.*
10. *Which are the most reliable frameworks for real-time classification?*

Bibliography

- [1] Wickline, S., Beyond intravascular imaging: Quantitative ultrasonic tissue characterization of vascular pathology, In: IEEE Ultrasonics symposium, 1994, pp. 1589–1597.
- [2] Arul, P. and Amin, V., Characterization of beef muscle tissue using texture analysis of ultrasonic images, In: Proceedings of the Twelfth Southern Biomedical Engineering Conference, 1993, pp. 141–143.
- [3] Mojsilovic, A. and Popovic, M., Analysis and characterization of myocardial tissue with the wavelet image extension [US images], In: Image Processing, 1995 (Proceedings) Vol. 2, pp. 23–26, 1995.
- [4] Jin, X. and Ong, S., Fractal characterization of kidney tissue sections, In: Engineering in Medicine and Biology Society, 1994. Engineering Advances: New Opportunities for Biomedical Engineers, Proceedings of the 16th Annual International Conference of the IEEE, Vol. 2, pp. 1136–1137, 1994.
- [5] Cohen, F. and Zhu, Q., Quantitative soft-tissue characterization in human organs using texture/attenuation models, In: Proceedings in Multidimensional Signal Processing Workshop, 1989, pp. 47–48.
- [6] Mavromatis, S. and Boi, J., Medical image segmentation using texture directional features, In: Engineering in Medicine and Biology Society, 2001. Proceedings of the 23rd Annual International Conference of the IEEE, Vol. 3, pp. 2673–2676, 2001.
- [7] Mavromatis, S., Mammographic mass classification using textural features and descriptive diagnostic data, In: Digital Signal Processing, DSP 2002. 14th International Conference on, Vol. 1, pp. 461–464, 2002.
- [8] Donohue, K. and Forsberg, F., Analysis and classification of tissue with scatterer structure templates, IEEE Trans. Ultrasonics, Ferroelect. Frequency Control, Vol. 46, No. 2, pp. 300–310, 1999.

- [9] Ravizza, P., Myocardial tissue characterization by means of nuclear magnetic resonance imaging, In: *Computers in Cardiology 1991 (Proceedings)*, pp. 501–504.
- [10] Vandenberg, J., Arterial imaging techniques and tissue characterization using fuzzy logic, In: *Proceedings of the 1994 Second Australian and New Zealand Conference on Intelligent Information Systems, 1994*, pp. 239–243.
- [11] Nailon, W. and McLaughlin, S., Comparative study of textural analysis techniques to characterize tissue from intravascular ultrasound, In: *Proceedings of the IEEE International Conference of Image Processing, Switzerland, IEEE Signal Processing Society, Piscataway, NJ, 1996*, pp. 303–305.
- [12] Nailon, W. and McLaughlin, S., Intravascular ultrasound image interpretation, In: *Proceedings of the International Conference on Pattern Recognition, Austria, IEEE Computer Society Press, Los Alamitos, CA, 1996*, pp. 503–506.
- [13] Nailon, W., Fractal texture analysis: An aid to tissue characterization with intravascular ultrasound, In: *Proceedings 19th International Conference, IEEE/EMBS, 1997*, pp. 534–537.
- [14] Spencer, T., Characterization of atherosclerotic plaque by spectral analysis of 30mhz intravascular ultrasound radio frequency data, In: *IEEE Ultrasonics Symposium, 1996*, pp. 1073–1076.
- [15] Dixon, K., Characterization of coronary plaque in intravascular ultrasound using histological correlation, In: *19th International Conference, IEEE/EMBS, pp. 530–533, 1997*.
- [16] Ahmed, M. and Leyman, A., Tissue characterization using radial transform and higher order statistics, In: *Nordic Signal Processing Symposium, 2000*, pp. 13–16.
- [17] de Korte, C. L. and van der Steen, A. F. W., Identification of atherosclerotic plaque components with intravascular ultrasound elastography in vivo: A yucatan pig study, *Circulation*, Vol. 105, No. 14, pp. 1627–1630, 2002.

- [18] Zhang, X. and Sonka, M., Tissue characterization in intravascular ultrasound images, *IEEE Trans. Med. Imaging*, Vol. 17, No. 6, pp. 889–899, 1998.
- [19] Pujol, O. and Radeva, P., Automatic segmentation of lumen in intravascular ultrasound images: An evaluation of texture feature extractors, In: *Proceedings for IBERAMIA*, 2002, pp. 159–168.
- [20] Pujol, O. and Radeva, P., Near real time plaque segmentation of ivus, In: *Proceedings of Computers in Cardiology*, 2003, pp. 159–168.
- [21] Randen, T. and Husoy, J. H., Filtering for texture classification: A comparative study, *Pattern Recogn.*, Vol. 21, No. 4, pp. 291–310, 1999.
- [22] Haralick, R., Shanmugam, K., and Dinstein, I., Textural features for image classification, *IEEE Trans. System, Man, Cybernetics*, Vol. 3, pp. 610–621, 1973.
- [23] Tuceryan, M., Moment based texture segmentation, *Pattern Recogn. Lett.*, Vol. 15, pp. 659–668, 1994.
- [24] Lindeberg, T., *Scale-Space Theory in Computer Vision*, Kluwer, Dordrecht, Netherlands, 1994.
- [25] Jain, A. and Farrokhnia, F., Unsupervised texture segmentation using gabor filters, In: *Systems, Man and Cybernetics*, 1990 (Conference Proceedings), pp. 14–19.
- [26] Mallat, S., A theory for multiresolution signal decomposition: The wavelet representation, *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 11, No. 7, pp. 674–694, 1989.
- [27] Mandelbrot, B., *The Fractal Geometry of Nature*, W. H. Freeman, New York, 1983.
- [28] Ojala, T., Pietikainen, M., and Maenpaa, T., Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 24, No. 7, pp. 971–987, 2002.

- [29] Julesz, B., Visual pattern discrimination, *IRE Trans. Inf. Theory*, Vol. IT-8, pp. 84–92, 1962.
- [30] Ohanian, P. and Dubes, R., Performance evaluation for four classes of textural features, *Pattern Recogn.*, Vol. 25, No. 8, pp. 819–833, 1992.
- [31] Martinez, J. and Thomas, F., Efficient computation of local geometric moments, *IEEE Trans. Image Process.*, Vol. 11, No. 9, pp. 1102–1111, 2002.
- [32] Caelli, T. and Oguztoreli, M. N., Some tasks and signal dependent rules for spatial vision, *Spatial Vision*, No. 2, pp. 295–315, 1987.
- [33] Chaudhuri, B. and Sarkar, N., Texture segmentation using fractal dimension, *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 17, No. 1, pp. 72–77, 1995.
- [34] Lindeberg, T., Scale-space theory: A basic tool for analysing structures at different scales, *J. Appl. Stat.*, Vol. 21, No. 2, pp. 225–270, 1994.
- [35] Rao, R. and Ballard, D., Natural basis functions and topographic memory for face recognition, In: *Proceedings of International Joint Conference on Artificial Intelligence*, 1995, pp. 10–17.
- [36] Lumbreras, F., Segmentation, Classification and Modelization of Textures by means of Multiresolution Descomposition Techniques, PhD Thesis, Computer Vision Center, Universitat Autònoma de Barcelona, 2001.
- [37] Jain, A. and Farrokhnia, F., A multi-channel filtering approach to texture segmentation, In: *Proceedings of Computer Vision and Pattern Recognition, CVPR'91*, 1991, pp. 364–370.
- [38] Fukunaga, K., *Introduction to Statistical Pattern Recognition*, Academic Press, New York, 1971.
- [39] Belhumeur, P., Eigenfaces vs fisherfaces: Recognition using class specific linear projection, *IEEE Pattern Anal. Machine Intell.*, Vol. 19, No. 7, pp. 711–720, 1997.

- [40] Schapire, R. E., The boosting approach to machine learning. An overview, In: MSRI Workshop on Nonlinear Estimation and Classification, 2002.
- [41] Viola, P. and Jones, M., Rapid object detection using a boosted cascade of simple features, In: Conference on Computer Vision and Pattern Recognition, 2001, pp. 511–518.
- [42] Duda, R. and Hart, P., Pattern Classification, Wiley InterScience, New York, 2001. Second Edition.
- [43] Sonka, M. and Zhang, X., Segmentation of intravascular ultrasound images: A knowledge-based approach, *IEEE Trans. Med. Imaging*, Vol. 17, No. 6, pp. 889–899, 1998.
- [44] von Birgelen, C., Computerized assessment of coronary lumen and atherosclerotic plaque dimensions in three-dimensional intravascular ultrasound correlated with histomorphometry, *Am. J. Cardiol.*, Vol. 78, pp. 1202–1209, 1996.
- [45] Klingensmith, J., Shekhar, R., and Vince, D., Evaluation of three-dimensional segmentation algorithms for identification of luminal and medial-adventitial borders in intravascular ultrasound images, *IEEE Trans. Med. Imaging*, Vol. 19, No. 10, pp. 996–1011, 2000.
- [46] McInerney, T. and Terzopoulos, D., Deformable models in medical images analysis: A survey, *Med. Image Anal.*, Vol. 1, No. 2, pp. 91–108, 1996.

Handbook of Biomedical Image Analysis

Volume 2: Segmentation Models Part B

Wilson, D.; Laxminarayan, S. (Eds.)

2005, XXII, 816 p. With CD-ROM., Hardcover

ISBN: 978-0-306-48605-0